RESEARCH ARTICLE

A formal model of neighborhood representation and applications in urban building aggregation supported by Delaunay triangulation

Tinghua Ai [ID]*, Hongmei Yin, Yilang Shen, Min Yang, Lu Wang

School of Resource and Environment Sciences, Wuhan University, Wuhan, China

* tinghuaai@whu.edu.cn

Abstract

Neighborhood relationship plays an important role in spatial analysis, map generalization, co-location data mining and other applications. From the perspective of computation, the formal model of neighborhood representation is a challenging question. This study presents a formal spatial data model for representing the planar spatial field with the support of Delaunay triangulation. Based on the three geometric elements in a triangle of the vertex, edge, and triangle area, the constructed data model describes the spatial objects of a point, line, and region respectively, as well as the neighborhood relationships among them. Three types of operators based on the model are formally defined, expanding, compressing and skeletonizing. For practical applications, three complex operators are extended by continuous and conditional operation. Through the application example of urban building generalization, this study illustrates the analysis of a neighborhood relationship and the detection of spatial conflicts, which is a crucial pre-process during map generalization. With the support of the proposed formal model of neighborhood representation, the generalization method uses the three basic operations of grouping, displacement and aggregation to perform decision making and detailed operation. The generalized result can retain the balance of built-up area better than that of other similar building generalization methods.

1. Introduction

Objects and phenomena in reality are represented as spatial entities and spatial relationships in a geographic information systems (GIS) conceptual world and are organized in spatial data models. Two spatial data models are commonly used: the object-oriented and field-oriented models. The object-oriented model emphasizes the integrity of entities in which the identified unit corresponds with the object in the real world, but it lacks an effective strategy for representing the inter-association or spatial relations, thus resulting in the requirement for complex vector computations. The field-oriented model focuses on the spatial correlation and continuity of the objects throughout the entire representing space. The surface is tessellated into sets

of units with the attribute description of each unit, and the relations between the entities can be obtained through set-based map algebra computation, such as the expanding of the raster. With respect to which is better in spatial representation, the corresponding argument between these two models is similar to that regarding waves and particles in the physics domain in the 20th century [1]. In the case of the field model, it is relatively easier to compute the spatial relationship and to conduct the spatial association processes. The map algebra method based on the raster data structure gives the field model advantage. Spatial associations, such as the neighborhood relationship, play an important role in spatial analysis, map generalization, co-location detection, and other applications. In map generalization, for example, the analysis of neighborhood relationships and detection of spatial conflicts usually comprises a significant pre-process. As the generalization attempts made to handle the spatial transformation from a large space to a small space, the spatial conflict detection within some neighboring areas is an important process. The neighborhood processing in the field model depends on the form of the data structure. For the regular data structure, such as the grid structure based on rectangles, equilateral triangle, and regular hexagons, we often use buffer operations to find the neighboring object within a certain distance. For an irregular data structure, such as the triangulated irregular network (TIN) structure, we can use the connection based on the Delaunay triangulation to find the neighboring objects. In this study, we attempt to explore the latter option by constructing a formal neighborhood model using the Delaunay triangulation and applying this in urban building generalization. In contrast to the application of TIN in the representation of the terrain surface in digital elevation model (DEM) studies, these models are applied to a planar condition with a focus on the neighborhood relation between buildings and, furthermore, on identifying spatial conflicts. However, these applications aim at different situations presenting concrete models associated with Delaunay triangulation without a unified formal model to represent neighborhood relationship. From the perspective of computation, the formal model is significant for describing the logical computation process. The formal definition and operators can be used to represent the neighborhood question in a normal and universal manner to support the algorithm design. In this study, we attempt to establish a formal model to represent a neighborhood relation based on the Delaunay triangulation to formally define spatial objects and present neighborhood-related operators.

We also attempt to apply the new model to the automatic generalization of an urban building cluster, which is associated with the neighborhood analysis. Automatic generalization of building clusters comprises different types of operations and analyses. First, it is necessary to group the buildings using different rules such as the Gestalt nature cognition or recognition of a distribution pattern. The displacement step is then used to identify how far and in which direction the objects should move. Third, the geometrical simplification and aggregation is conducted to abstract the building shape. The three steps require a model related to the neighborhood analysis to derive such descriptive indices for the distribution pattern, distribution density, adjacency direction, adjacency distance, etc.

In building data generalization, the operation can be classified into two types according to the object type, that is individual building generalization and cluster building generalization. Several works have developed algorithms and methods for independent building simplification [2–18]. Regnauld and Edwardes [2] proposed three typical operations for the simplification of buildings from the perspective of the readable view: squaring, detail removal, and local enlargement. By separating a building into some hierarchical rectangle elements, Guo and Ai [4] presented a method to simplify the building polygon based on an idea of divide-and-conquer. By treating the four continuous adjacent points as an overall unit, Xu et al. [7] proposed an approach for building simplification based on the improvement of local simplification algorithms. In this method, the bend structures can be analyzed and the short edges can be

removed. Sester [13] proposed an approach of building simplification comprising the use of a set of rules and adjusted the simplified building optimally based on the theory of least squares adjustment. In this method, the short edges can be eliminated using an optimization solution. In terms of artificial intelligence, Cheng et al. [17] presented a model of back-propagation neural network model for learning cartographers' knowledge for the realization of building simplification. However, these studies usually only took into consideration the geometric features and ignored the neighborhood relationships of the building during the generalization.

As compared with the individual building simplification, the building cluster generalization is more complex and requires the consideration of a greater number of aspects, such as the group pattern detection, context analysis, and spatial conflict judgment. An efficient model is thus required for the neighborhood analysis. Regnauld [19] presented an approach for classifying building groups using the multiple spanning tree model in graph theory. Li et al. [20] proposed an approach for group building cluster that combined the Gestalt theory and urban morphology. Chaudhry and Mackaness [21] combined the buffer of building objects and the derivation of the single building object surrounding clusters for the generalization of city boundaries. Shen [22] et al. applied the superpixel segmentation technologies for building aggregation. Proximity analysis plays a very important role in building generalization because displacement conflicts or topology errors may be produced during the generalization. Thus, we attempt to apply the proposed neighborhood representation model to the automatic generalization of urban building clusters. An outstanding property exists in our neighborhood analysis model, in that, it can simultaneously support several operations including grouping detection, displacement, and aggregation. The minimum spanning tree (MST) model in Regnauld's method [19] and the urban morphology model in [20] can be used to conduct only one operation aggregation. The difference between our method and the others is the change in the built-up area after the map generalization. Our proposed method is effective in balancing the built-up area as the displacement prevents the gap area from being absorbed into built-up area.

This paper is organized as follows: the neighborhood representation model of the spatial field is discussed in Section 2. Section 3 presents the expanding and compressing operation based on the proposed formal triangulation data model (FTDM). A progressive algorithm of building cluster aggregation based on FTDM is presented in Section 4 with experiment illustrations, and some future improvements are discussed in the conclusion in Section 5.

2. Neighborhood representation model based on Delaunay triangulation

In the field-oriented model, the TIN structure covers the whole region without overlaps or gaps between the tessellated triangles. The relations among spatial entities are represented by the connection of triangles. As a special geometrical construction to support the TIN establishment, the Delaunay triangulation has the characteristics of "circumcircle rule" and "maximum rule of the smallest angle," and these properties make it an efficient model for representing the spatial neighborhood relationship [23, 24]. In the domain of map generalization, the Delaunay triangulation and its dual Voronoi diagram are widely used in the detection of conflict and neighbor relationships between objects, thus generating various data models aimed at different purposes. For example, Jones and Ware [25, 26] built the simplicial data structure (SDS) model for searching for area cluster targets and merging the adjacent objectives; the enhanced formal data structure (EFDS) model developed by Peng [27] was an improved model of the formal data structure (FDS) model developed by Molenaar, is which focused on the extraction of a "safe area" and "non-safe area" in the target generation space. Other works in which the

Delaunay triangulation was applied to map generalization include the building pattern recognition before generalization [28], the displacement of crowded building and conflicted point symbol [29, 30, 31], line simplification [32–35], polygon decomposition during land-use data generalization [36], and watershed area computation in hydrographic network generalization [37]. These properties make the Delaunay triangulation an efficient tool in GIS modelling applications. One such typical application is the representation of the terrain surface by TIN model. However, the aforementioned property of the Delaunay triangulation also makes it an efficient tool for neighborhood representation in a planar space, as shown in Fig 1. Through the triangle connection, the neighborhood relation among the spatial objects in the two-dimensional space can be detected. In this section, we attempt to build a formal model of neighborhood representation using the Delaunay triangulation.

2.1. Formal definitions

The spatial objects including points, lines, and polygons must be embedded in the Delaunay triangulation and then through the triangle edges to detect the neighborhood relation. We build a model to involve all the spatial objects to be studied and use the vertices of linear objects and area objects be a point set for the triangulation construction.

We first determine the geographical extent of the spatial representation. We then add three points Q_1, Q_2, and Q_3 to ensure that all the spatial objects are within the triangle identified by the three added points. The Euclidean space containing $\Delta Q_1 Q_2 Q_3$ is denoted as Δ. We use the planar points to construct the constrained Delaunay triangulation, which is expressed as $W\langle V, E, T\rangle$, where $V = \{v_1, v_2, \cdots, v_m\}$ is a non-empty point set, $E = \{e_1, e_2, \cdots, e_n\}$ is a non-empty edge set, and $T = \{t_1, t_2, \cdots, t_n\}$ is a non-empty triangle set.

Obviously, W is a graph $G\langle V, E\rangle$ with V as the vertex set and E as the edge set. The edge $e_i \in E$, is represented by a point pair comprising the points of set V as $e_i(v_{i0}, v_{i1})$. The triangle $t_i \in T$, comprises the three edges of set E and is represented as $t_i(e_{i1}, e_{i2}, e_{i3})$. The triangle set T covers the entire domain space. For any $t_i, t_j \in T$, there is $t_i \cap t_j = $ Null or DIM $(t_i \cap t_j) = 1$, where the latter means that the two triangles are adjacent and the dimension of the intersect region is 1, i.e., a line. When there is a unique sharing edge between triangles t_i and t_j, namely, $\exists u, v: e_{iu} = e_{jv}$, triangles t_i and t_j are called adjacent.
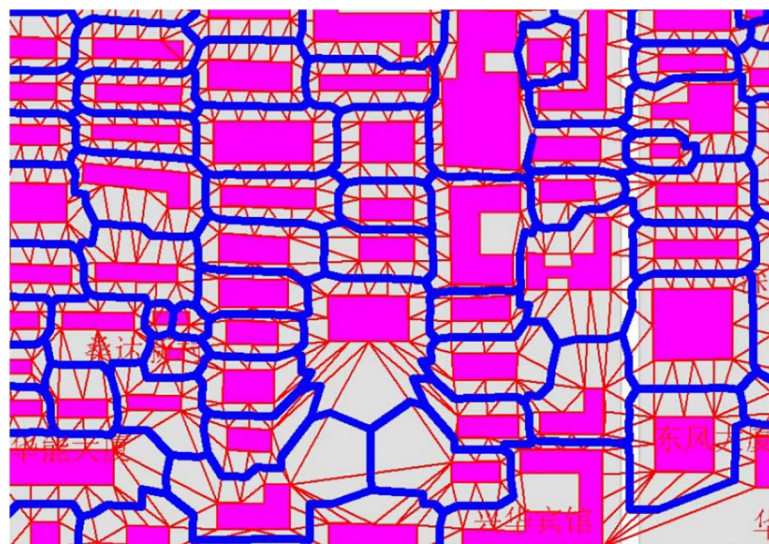

Fig 1. Detecting the neighboring region among building cluster using Delaunay triangulation.

We define the functions related to the neighborhood analysis as follows:

Neighbor(t_i,e_{ij}): $T \times E \rightarrow T$ returns the triangle t_j that is adjacent to triangle t_i and the sharing edge is e_{ij}. As shown in Fig 2, the triangle t_i and t_j has the sharing edge e_{ij}.

Neighbors(t_i): $T \rightarrow T$ returns the set of triangles that are adjacent to t_i, and:

Neighbors(t_i) = {*Neighbor(t_i, e_{i1})*, *Neighbor(t_i, e_{i2})*, *Neighbor(t_i, e_{i3})*}. As shown in Fig 2, the neighboring triangles of t_i are *Neighbors(t_i)* = {t_j,t_o,t_s}.

Joins(v_i): $V \rightarrow T$ returns the set of triangles that shares the vertex v_i. In Fig 2, *Joins(v_i)* = {t_i,t_j,t_p, t_q,t_o}.

Condition(c): {logic expression c}$\rightarrow T$ returns the triangle set T_c that satisfies condition c.

Access(t_i,t_j): $T \times T \rightarrow$ {TRUE, FALSE}, when there exists one set of edges between any vertex of t_i and vertex of t_j in graph $G\langle V, E\rangle$, returns TRUE or else returns FALSE. In Fig 2, *Access (t_p, t_s)* = *true*, as there is path connecting triangle t_p and t_s.

Begin(e): $E \rightarrow V$ returns the start point v_b of edge e.

End(e): $E \rightarrow V$ returns the end point v_e of edge e.

2.2. Representation of spatial objects

We denote the set including all the objects within the space as F_Δ, which includes all the point objects P_Δ, all linear objects L_Δ, and all area objects A_Δ. We then have $F_\Delta = P_\Delta \cup L_\Delta \cup A_\Delta$. We now define the triangulation model representing space objects called FTDM for short.

Point object: $p \in P_\Delta$ is defined such that if the element v_i belongs to V, then for any point p belonging to $W\langle V,E,T\rangle$ there is $p \in V$.

Line object: $l \in L_\Delta$ comprises a series of line segments. In $W\langle V,E,T\rangle$, these segments must belong to E. l can be defined as: the ordered subset {e_0,e_1,$\cdots$,e_n} in E that satisfies the condition $end(e_i) = begin(e_{i+1})$ for any $i \in [0,n-1]$.
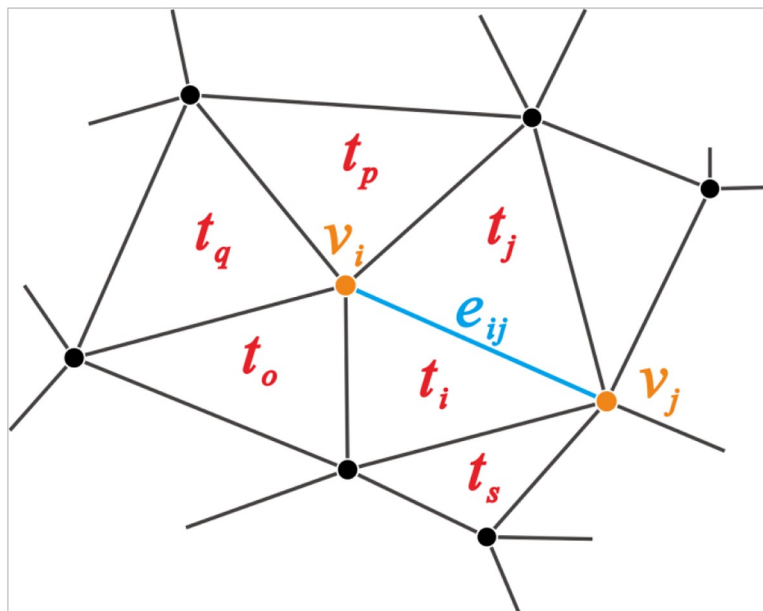

Fig 2. Formal definitions of the functions related to neighborhood analysis.
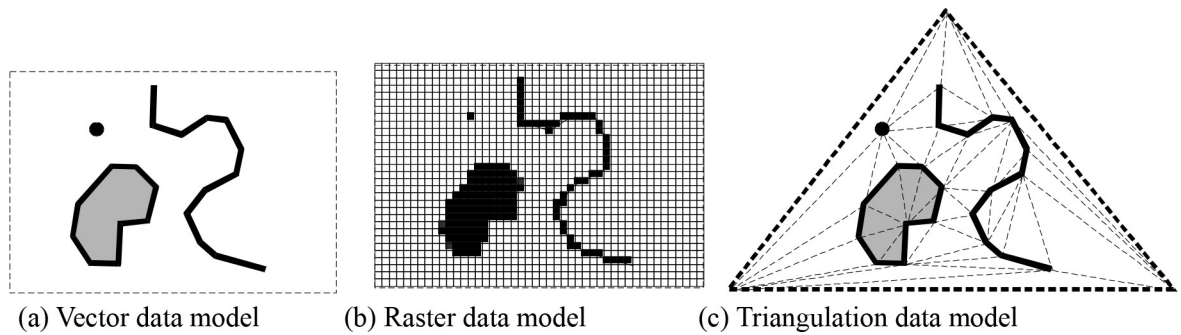
(a) Vector data model　　(b) Raster data model　　(c) Triangulation data model

Fig 3. The representation of spatial objects in three data models.

Area object: $a \in A_\Delta$ can be defined as the subset $\{t_0, t_1, \cdots, t_m\}$ that satisfies the condition Access (t_i, t_j) = TRUE for any $i, j \in [0, m]$.

Fig 3 shows that the representation of various spatial objects in the FTDM model, which is obviously different from that in the raster model, which uses a regular grid unit to represent a point, line, and region. The FTDM model comprises three elements: a point unit, boundary unit, and triangle unit, while there is only one structural unit in the raster model. Triangles in the FTDM model play two roles: the component of area object and the connection between objects. The triangles playing the connecting role can connect objects according to "visibility" between the objects and "the nearest link" based on the Delaunay triangulation, irrespective of the geometric distance between the objects. However, in the raster model, the connection between objects is usually formed by the ordinal linkage of a number of structural units. When searching for the neighboring object, the raster calculation is necessary for obtaining the results, and the four- or eight-direction expansion in the raster calculation is not only blind, but also has no neighboring connecting properties of triangulation.

3. Neighborhood operators based on the FTDM model

3.1. Simple operators

For simulating the operation of the raster model, two types of operators of FTDM are defined based on the constrained Delaunay triangulation $W\langle V, E, T \rangle$: expanding and compressing. Firstly, we define the region of the triangle set and related concepts. Region r is defined as the subset of T that satisfies certain conditions, and we let r be the single connected region, i.e., $r = Condition(c) \subset T$. The boundary b of region r is represented as follows:

$$b = Boudary(r) = \{e_i | \exists t : t \in Condition(c) \wedge Neighbor(t, e_i) \notin Condition(c)\}$$

The boundary defined here is the set of the edges of the triangles, without considering the situation that organized them in an orderly manner to form a closed ring.

Based on boundary b, we denote the outside neighbor of the boundary b_{out} and the inside neighbor of the boundary b_{in} as follows:

$$b_{in} = \{t_i | t_i \in r \wedge (\exists e : e \in b \wedge (t_i \in Joins(begin(e)) \vee t_i \in Joins(End(e))))\}$$

$$b_{out} = \{t_i | t_i \notin r \wedge (\exists e : e \in b \wedge (t_i \in Joins(begin(e)) \vee t_i \in Joins(End(e))))\}$$

Based on b_{in} and b_{out}, we define the regional expanding operator $Expand()$ and regional compressing operator $Compress()$ as follows:

Expand(*r*):*T*→*T*, the transformation of region *r* that meets condition *c*, the return value is $r \cup b_{out}$.

Compress(*r*):*T*→*T*, the transformation of region *r* that meets condition *c*, and the return value is $r - b_{in}$.

The transformation result of the two operators is presented in Fig 4.

3.2. Complex operators

The expanding and compressing operations can be applied to the region iteratively. We introduce the continuously expanding operator denoted as $Expand^{(n)}(r)$, and the continuously compressing operator denoted as $Compress^{(n)}(r)$. The recursive definition for $Expand^{(n)}(r)$ is

$$Expand^{(0)}(r) = r;$$

$$Expand^{(n)}(r) = Expand(Expand^{(n-1)}(r)).$$

The recursive definition for $Compress^{(n)}(r)$ is

$$Compress^{(0)}(r) = r$$

$$Compress^{(n)}(r) = Compress(Compress^{(n-1)}(r));$$

$Compress^{(n)}(r)$ results in the repetitive peeling of the region, and $Expand^{(n)}(r)$ results in the gradual expansion of the region.

The operator *Expand*(*r*) and *Compress*(*r*) for the region $r \in T$ is non-conditional, but for some operation in the real application, it is necessary to perform conditional filtering for the expanded triangular set b_{out} and the eliminated triangular set b_{in} of *Expand* and *Compress*, respectively. Therefore, we further define the conditional expanding operator $Expand_c(r)$ and the conditional compressing operator $Compress_c(r)$.

Let us suppose the condition is c_0, then

$$Expand_c(r) : T \rightarrow T, \text{return } r \cup (b_{out} \cap Condition(c_0));$$

$$Compress_c(r) : T \rightarrow T, \text{return } r - (b_{in} \cap Condition(c_0));$$

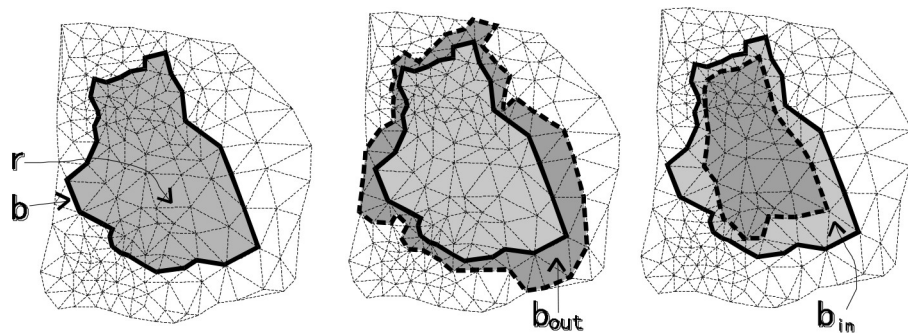The return value can be several non-connected subsets of triangles.

Fig 4. The illustrations of region expanding and compressing. *r* denotes the region, *b* the boundary, b_{in} the inside neighbor, and b_{out} the outside neighbor.

The introduction of conditional restriction in the continuous operator results in a conditional continuous operator, which is denoted as $Expand_c^{(n)}(r)$ and $Compress_c^{(n)}(r)$, respectively. The conditions for $Expand_c^{(n)}(r)$ and $Compress_c^{(n)}(r)$ can be classified as follows.

(1) Target achieved condition

In the expanding or compressing operation, when the expanded or eliminated triangle has a certain relation with the spatial target, then the operation is stopped. This relation includes when the vertex of the triangle reaches the appointed point target, the edge of the triangle is attached to any edge of the line target or area target, or the polygon is attached to any area target. Under this condition, the neighbor relationship between objects can be determined via two operations.

(2) Geometric feature condition

When the perimeter of the triangle, area of the triangle, length of the arc, or area of the closed ring clipped by the edge of the triangle on the edge of the polygon meet the threshold conditions, then the operation is stopped. The triangles in the triangulation $W\langle V,E,T\rangle$ are merely connected based on the visibility between targets. The spatial distance should be considered when judging the contiguity of the triangles, and there should be some restrictions on the length of their edges.

(3) Spatial measurement condition

During the procedure of expanding or compressing, we consider the distance or orientation from the edge of the region, center of the region, or other reference locations as the restrictions. This is suitable for the operation on the anisotropy spatial field. Weights should be added to the expanse of the operation in different directions.

(4) Neighboring freedom condition

The neighboring expanding in a raster model comprises two cases: eight-direction and four-direction expansion. Similarly, the operation in the FTDM model has edge neighboring expanding and point neighboring expanding. In the above research, the application of function $Joins(v_i)$ is point neighboring muti-direction freedom. When the adjacent connection for triangles is restricted, the freedom is three.

From the practical judgment point of view, the above conditions can be determined using two classes. One class is based on qualitative judgment for selecting the conditional region in which the triangle touches certain types of objects or with a connection relation to certain object. The other class is based on quantitative measurement for selecting the conditional region in which the triangle area, edge length, and movement angle exceed a pre-defined tolerance, such as the area of 4 mm^2, distance of 2 mm, and the angle of 0.5 π. In the following section 4, an example will be presented to illustrate the continuous conditional operator $Expand_c^{(n)}(r)$ through a distance measure to determine how to expand and then finally stop.

It is worth noting that although the operators of $Expand(r)$ and $Compress(r)$ have opposite effects on the change in the region extending range, they do not generally satisfy the equation: $r = Compress(Exprend(r))$ or $r = Expand(Compress(r))$. This is because the restrictions in the expansion and compression procedure vary in different directions, and it thus is impossible to achieve an isotropic effect.

3.3. Skeletonizing operator

The use of a refinement operator in the raster model is one method for converting the raster into the vector skeleton. In the FTDM model, we can also define the skeletonizing operator $Skeleton(r)$, which is a function that converts region r into the graph structure related to a skeleton structure.

We first classify the triangles in the conditional region $r = Condition(c)$ into three types. We define function $f(t)$ as $Condition(c) \rightarrow \{0,1,2,3\}$. The return value is the number of elements in the set of $Neighbors(t) \cap Condition(c)$, which is denoted as $f(t) = \|Neighbors(t) \cap Condition(c)\|$. We classify the triangles in r into three types: if $f(t) = 1$, then t belongs to Type I; if $f(t) = 2$, then t belongs to Type II; if $f(t) = 3$, then t belongs to Type III; and if $f(t) = 0$, then t is isolated triangle, out of consideration. To facilitate the narrative, we call the shared edge between t and $Neighbors(t)$ as the neighbor edge.

We now define the regional skeletonizing operator $Skeleton(r)$. Let us suppose that the set of all graphs is G_Δ, then $Skeleton(r): T \rightarrow G_\Delta$, the return value is $G\langle E',V'\rangle$, where E' and V' are the generated set of edges and the set of vertices, using the skeletonizing method, respectively.

As shown in Fig 5, for Type I triangles, we connect the mid-point of the unique neighbor edge with the corresponding vertex; for Type II triangles, we connect the mid-points of the neighbor edges; while for Type III triangles, we connect the center of gravity with the mid-points of all the edges.

$$\text{Type I}: \ A \rightarrow P_1 \text{ or } P_1 \rightarrow A;$$

$$\text{Type II}: \ P_1 \rightarrow P_2 \text{ or } P_2 \rightarrow P_1;$$

$$\text{Type III}: \ O \rightarrow P_i \text{ or } P_i \rightarrow O, \ i = 1, 2, 3.$$

Starting from Type I or Type III triangles, searching along the neighbor relationship across Type II triangles and ending with Type I or Type III triangles, we can obtain one edge of the graph formed by a series of points $\{Q_i\}$. The construction of the graph is finished when all the Type I triangles have been searched once, and all the Type III triangles have been searched three times.

For the extracted graph structure, as shown in Fig 6, each edge (indicated by a red line) corresponds to a skeleton between two boundaries of two objects B_1 and B_2 on two sides. This represents the skeleton across a set of triangles (indicated in green) connects two objects (B_1 and B_2 in Fig 6).

We can compute the distance between two objects based on the average height of the triangles crossed by skeletons. As shown in Fig 7, we scan all the triangles across the skeleton. We compute the local distance for each triangle and then integrate the weighted distance as the average distance between two objects. For three types of triangles, the corresponding local
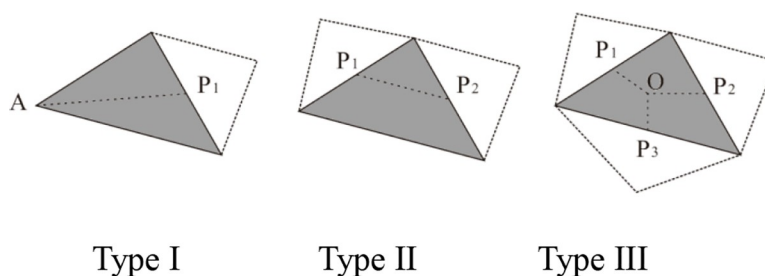

Type I   Type II   Type III

Fig 5. The three types of triangle and the skeleton line connection for each one.
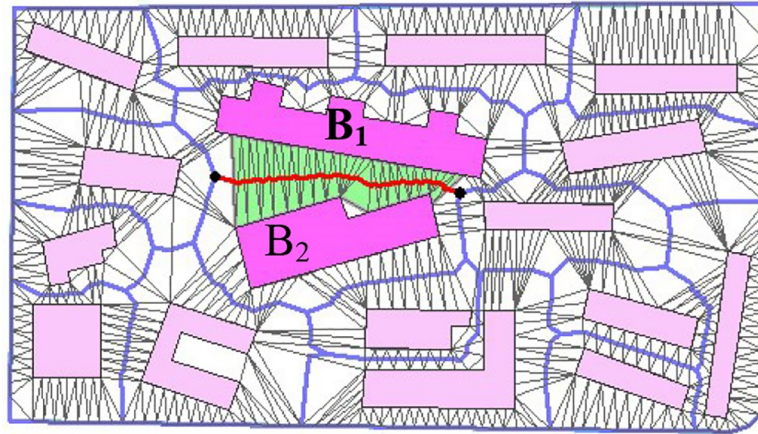
Fig 6. The extraction of the skeleton from the triangle region.

distance W_1W_2 is presented in Fig 7. The computation function of the average distance $\hat{w}$ is

$$\hat{w} = \sum_{i=0}^{k} \frac{\|Q_i Q_{i+1}\|}{l} \|W_{i1}W_{i2}\|$$

where l is the length of the entire skeleton, and k is the number of triangle involved. $\hat{w}$ is also called the skeleton width. This weighted distance computation based on the skeleton takes into consideration the building shape structure, spatial distribution, and other building's influence.

The skeleton operator can obtain the main central line of a region, such as a polygon. In the field of geometric shape analysis, there also exist other methods for extracting the polygon central line, for example, the medial axis transportation (MAT) based on the angular bisector operation [38], the superpixel segmentation method [39]. We use the same polygon data with a complex shape for conducting the comparison between the MAT-based method and the Delaunay-triangulation-based method. From the result of the example that is presented in Fig 8, it can be observed that the Delaunay-triangulation-based method has fewer hair segments, and the skeleton shape is smoother than that of the MAT-based method. An in-depth comparison of these two methods is outside the scope of this study, while a detailed discussion of the different methods can be referred to in [4] and [38].

The operator skeletonizing is performed on the region covered by the triangle set. We use different condition c to extract a subset of triangles, and the skeletonizing results in different graphs for representing the corresponding geometric structures. For a polygon cluster, if we extract the triangle region within the polygon, the skeletonizing will obtain the medial structure line as shown in Fig 9 (left). This process can be used to simplify the hydrographic features
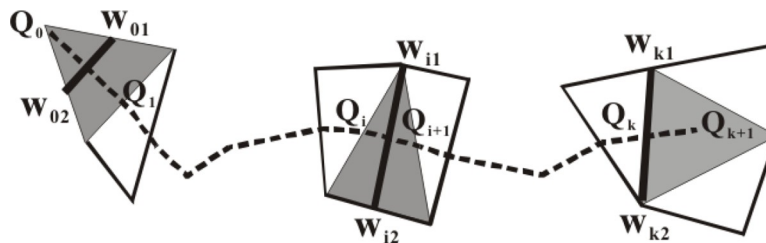

Fig 7. The local distance representation of $W_{i1}W_{i2}$ for three types of triangles.
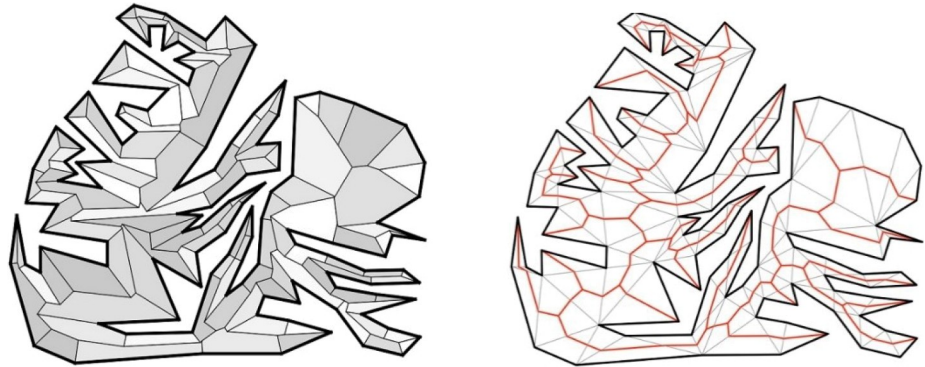
Fig 8. A comparison between the MAT-based skeleton (left) and Delaunay-triangulation-network-based (DTN-based) skeleton (right) using the same complex polygon data.

by collapsing narrow polygons into a line representation. If we extract the triangle region of the street roads (outside a street block), the skeletonizing provides the medial line of the street features for supporting applications such as navigation. If we extract the triangulated region outside the building polygons, the skeletonizing provides a similar Voronoi diagram to tessellate the polygon cluster, in which the neighboring polygon is separated by the connected skeleton line, as shown in Fig 9 (right).

4. Application of FTDM model in building cluster generalization

The building or built-up area in an urban region has special properties with respect to distribution structure, cluster pattern, and alignment shape. In spatial data scaling, the generalization of building data within street block is an attractive question requiring special strategies to consider the aforementioned properties [40]. Some methods have been developed to settle the abstraction of a building within a street block, such as the aggregation under the control of Gestalt principles and urban morphology [20], the application of an optimization method in building cluster abstraction [12], and using the MST model to group building clusters [19]. These methods can be used to conduct only one operation: either building group detection or neighbor building aggregation. The resulting issue is that the aggregated building will greatly
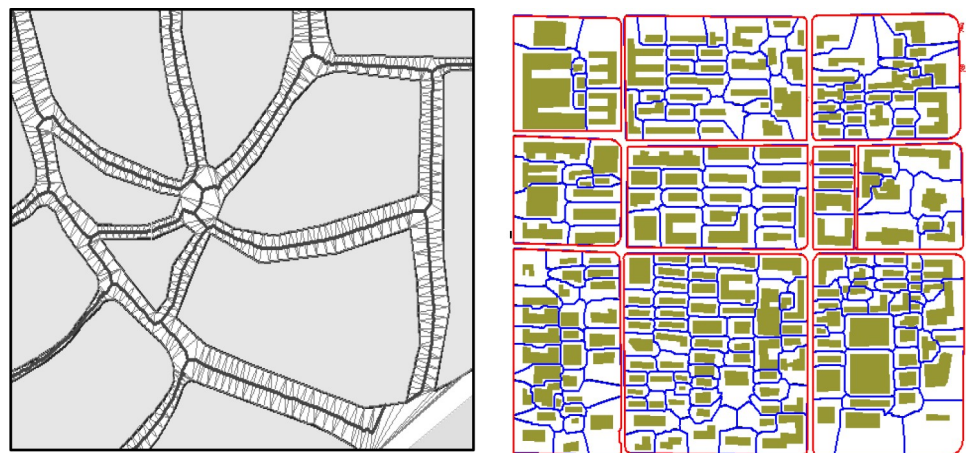

Fig 9. (Left) performing skeletonizing outside street blocks results in the medial street line, (Right) performing skeletonizing on building clusters within a street block.

increase the built-up area as the gap area between the neighbors has been included in the built-up region. With the objective of resolving this issue, we develop a method based on the FTDM model to perform building generalization to maintain the balance of the built-up area as well as possible. We use the FTDM model to detect the competition area and perform displacement and aggregation together to prevent the gap area from becoming absorbed into built-up area.

We define the building cluster generalization such that it includes four steps: grouping, displacement, aggregation, and simplification. The grouping step comprises the detection of the cluster pattern, which should be preserved after abstraction. The displacement operation attempts to move buildings within a group closer to avoid increasing the built-up area. The aggregation operation attempts to combine the closed building group into one block. The final simplification step is used to make the shape of the built-up area simple with orthogonal characteristics. The proposed FTDM model can be used to support the generalization of urban building clusters in several aspects. Through triangle tessellation and neighborhood analysis by triangle expansion, we can detect the building group and conflict area and further through skeletonizing operations, move closed buildings together. It is the displacement that guarantees the balance of the built-up area after the building data generalization. In this section, we present the application of the FTDM model in building data generalization and focus on three operations: grouping, displacement, and aggregation.

The entire process of the building data generalization is based on the FTDM model, comprises nine steps, and is illustrated in Fig 10. The first constructed triangle set covers the whole building area and obtains the tessellation as the FTDM model to support the next neighborhood analysis. We then use the formal operations *Expand(r)* and *Skeleton(r)* in the FTDM model to extract the building group and further to move the closed building together. Finally, the aggregation of the moved neighbors provides the generalization result. In the nine steps of the building generalization, the formal operator *Expand(r)* and *Skeleton(r)* in the FTDM model plays an important role.

4.1 Grouping and conflict detection

As shown in Fig 10 (from step 1 to 5), after the construction of the building cluster triangulation we obtain the FTDM model. For any building object, we select one triangle (indicated in red or green in Fig 10) touching it as the expansion seed. Then, we perform $Expand_c^{(n)}(r)$. This is a continuous and conditional expansion operation (see its definition and performing function in section 3). The condition here is that the triangle should be located outside the building polygon, and the length of the triangle edge between the neighboring buildings should be less than the pre-defined tolerance d. The value of d depends on the neighbor length according to the aggregation gap distance, such as 2 mm in paper space. In Fig 10, steps 2-4 complete the continuous and conditional expansion, and the operation is stopped when no boundary triangle satisfies the expansion conditions. After performing step 4, we obtain two regions of the triangle set indicated by light red and light green, respectively. The building object associated with the triangle set can be assigned to one group.

Before step 5 in Fig 10, we observe that one building group is connected by a set of triangles locating in the gap area. The narrow region with a gap distance of less than tolerance d has been detected. For each set of triangles connecting gap area, we perform the operation *skeleton (r)* and obtain the skeleton central line between the neighboring buildings as shown in step 5. We define the skeleton using a gap distance of less than the tolerance d the conflict skeleton. An example of the detection of the conflict skeleton and conflict building is illustrated in Fig 11.
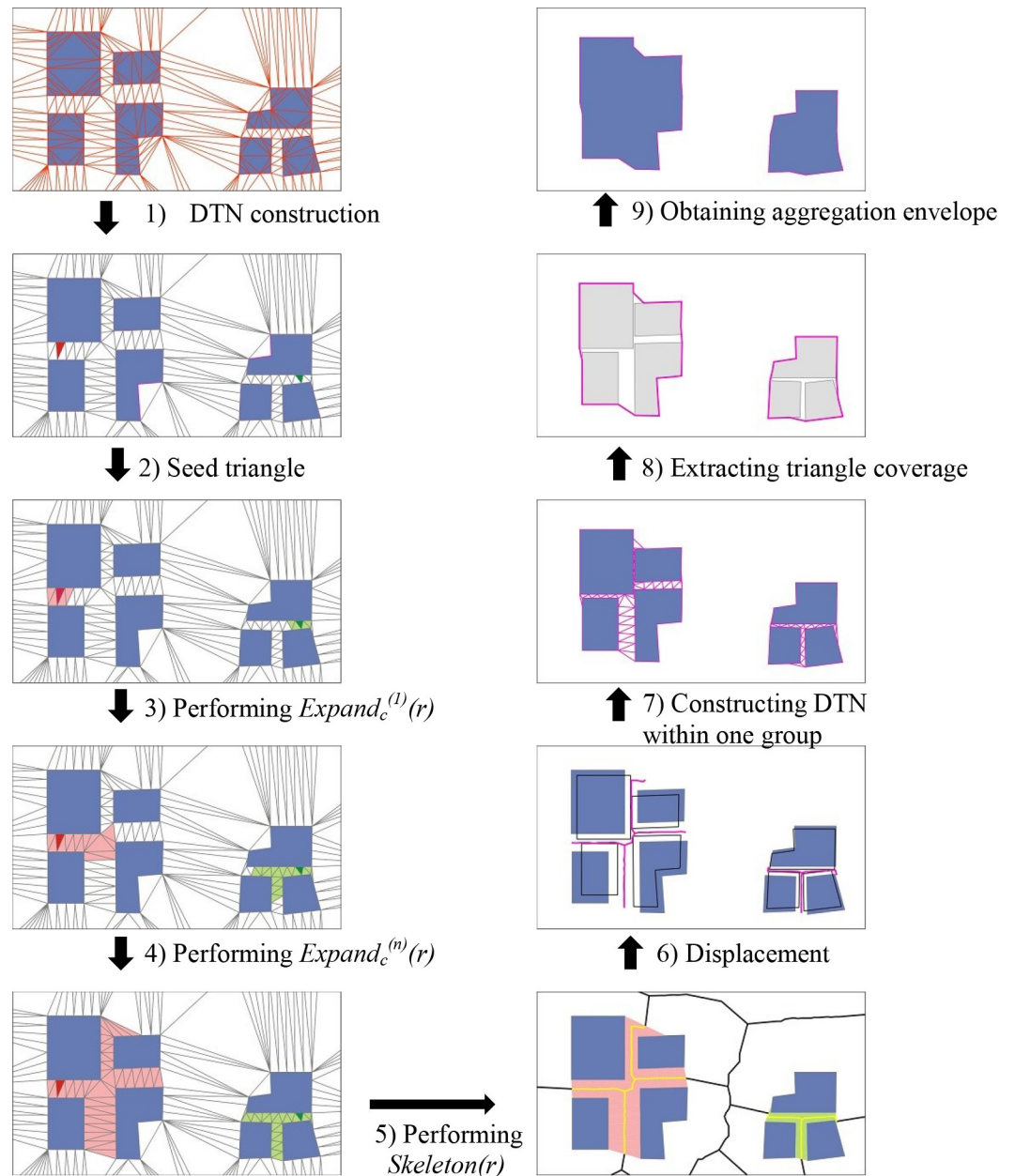
Fig 10. The illustration of the building aggregation using FTDM and operators *Expand(r)*, *Skeleton(r)*.

https://doi.org/10.1371/journal.pone.0218877.g010

4.2 Displacement within one group

The building facing the conflict skeleton is called the conflict building. The neighboring closed conflict buildings move together before the aggregation as shown in step 6 of Fig 10. The displacement in the building cluster generalization is aimed at keeping built-up area balanced.

The analysis of the conflict buildings offers an answer to the question of which one will be displaced in later generalization. How far, and in which direction, the conflict building will move are to be determined next. The normal direction of the line of the conflict skeleton fitted by the least square method will be the moving direction, as indicated by the arrow symbol in Fig 11. When the conflict object has only one conflict skeleton, the moving direction is

Fig 11. An illustration of the conflict skeletons, conflict OPs (Object Polygon) visualized as red lines, and the arrows represent the building movement direction.

determined. Else, the integrated moving direction can be calculated by using the vector sum by the parallelogram rule. It is estimated that neighbor conflict attracts each conflict building with the same attraction force. One building will stay unchanged if it is attracted by neighbors from two opposite directions or is surrounded by conflict buildings (i.e., all skeletons related to one building are conflicted by each other). In a practical application, it can be considered that no one direction attraction is stronger than any other direction and that the object is fixed if the length of the sum vector is shorter than a threshold. The movement direction of the conflict object is illustrated in Fig 11 by a dark arrow symbol representing the displacement direction and a dark dot representing the fixed building.

In terms of the offset length of the displacement, it is assumed that the position accuracy is at least half the conflict distance, which means that the in-face movement and the meeting in one position of the conflict building are within the position accuracy. We draw extended lines that are parallel with the displacement direction, from each vertex of the conflict building and then compute the distance between the start vertex and the intersection point of the extended line and skeleton. The shortest one is set as the displacement offset length. It can be guaranteed that the movement does not result in the building crossing the envelope skeleton or its overlapping with other neighboring buildings. No other new conflicts are produced from this type of displacement; in displacement generalization research, this is essential.

However, generally speaking, after the above displacement, it is not guaranteed that two buildings will share exactly the same boundary seamlessly. Usually, small gap areas still exist. One possible solution to this issue is applying a rotation to it, even though it is difficult and complicated to decide the angle and range of rotation and solve the problem perfectly.

4.3 Aggregation of neighboring buildings

Steps 7 to 9 in Fig 10 illustrate the aggregation of neighboring buildings within one group. After the displacement of conflict building together, however, there still are small gap areas among the buildings [41]. It is necessary to construct an envelope polygon to cover all the buildings within one group. A simple method to achieve this is to use a basic operations buffer

and overlap in common GIS functions. We first use an expansion buffer operation to exaggerate each building area, then use union overlap to obtain the coverage polygon, and finally perform a compression buffer to shrink back. The obtained result is an aggregated polygon. In our experiment, we apply the method based on the Delaunay triangulation again. Through the close connection with the Delaunay triangulation to extract the envelope polygon. This method has been described in the SDS model by Jones and Ware [25, 26].

4.4. Progressive generalization process

The group detection and detailed geometric operations in building data generalization is discussed above in Fig 10. The original data can be required from this website: https://pan.baidu.com/s/1Zr_H9cvdEvX3QzOhK3ASXwv (extracting code: jnxx). For the whole working process, it is necessary to organize the operators using some parameter control. Let us consider the case in which conflicting building object associate with each other. By moving and aggregating one conflict building into its neighbor, the conflict on the neighboring location may be also eliminated. The conflict among the parts can be solved by the displacement and aggregation of partly conflicted objects. Thus, we can use a progressive strategy to aggregate the conflict buildings. The progressive generalization process is described as follows.

We repeat the following steps until no conflict is found:

1. (1) Construct DTN triangulation, and find conflict skeleton and conflict building object based on FTDM model.

2. (2) Classify buildings into different groups according to the conflict skeleton connection.

3. (3) Scan the building group and use the method in Fig 10 to aggregate the closed neighboring buildings.

4. (4) Eliminate the remaining conflicts after the aggregation.

Fig 12 presents some steps of the building cluster generalization and the generalized results as well as tessellation results. If the building is distributed in a common situation without a too-crowded distribution, a general appropriate outcome can be obtained using the above working process. As can be observed from the regions indicated by the red box in Fig 12, the
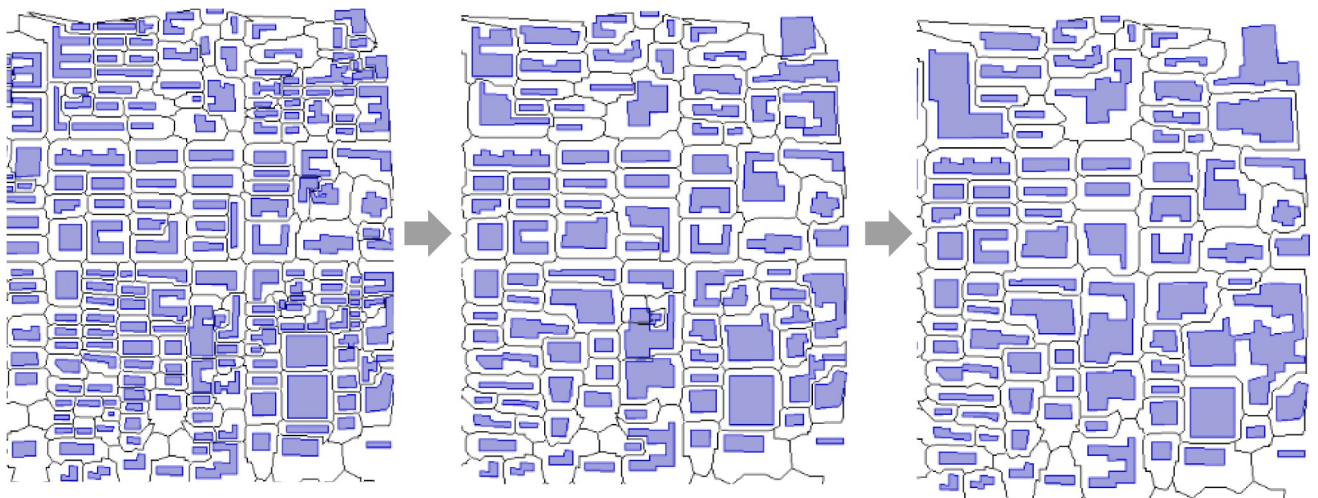

Fig 12. The progressive generalization of a building cluster based on the FTDM model.

original buildings are reasonably aggregated with the increase proximity distance, and the results are visually satisfactory. However, sometimes the location of the early aggregated building may produce a slight displacement as the skeleton is changed gradually throughout the process. In this method, the degree of generalization depends on the definition of the tolerant conflict distance.

4.5. Experiment discussion

Based on the FTDM model and formal operations *Expand(r)* and *Skeleton(r)*, we develop the method of building data generalization, which takes into consideration three operations in generalization, that is, grouping, displacement, and aggregation (the other operation of simplification of the building shape is not involved in this study). The advantage of this generalization is that the displacement before the aggregation maintain the balance of the built-up area as well as possible. The property of the remaining built-up area without a large change is an important requirement in map generalization [42, 43]. In this proposed method, the neighboring buildings move toward to each other and eliminate the gap area between them as well as possible. However, this cannot guarantee that two buildings share exactly the same boundary seamlessly. Small gap areas still exist, but the involved gap area reduces greatly. Some other aggregation methods, such as those used in [25, 20], aggregate the neighboring objects directly, thus resulting in a significant increase in the built-up area. We conduct an experiment of building aggregation based on the same data using the aggregation function in ArcGIS and our proposed method respectively. The comparison between the two methods can be made observed in Fig 13 and Table 1.

The left side of in Fig 13 illustrates the aggregation result and the skeleton tessellation obtained using our method, the middle the aggregated result overlapping with original building data, and the right the same operation using the ArcGIS function. In Fig 13B, the buildings after aggregation using the proposed method are not inclined to adhere to the original buildings. However, the aggregated buildings using the ArcGIS method are inclined to adhere to the original buildings, as shown in Fig 13C. In ArcGIS ToolBox, we apply the command "Aggregate Polygons" the function of which is to aggregate the neighboring polygons. We set the tolerant gap distance $d = 15$ m for both the methods. From the comparison, we can observe that the aggregation performed using the ArcGIS function includes gap areas and greatly increases the built-up area. In Table 1, we find that the proposed method increases the built-up area by 5.1%, which is less than 17.4% for the ArcGIS aggregation method.
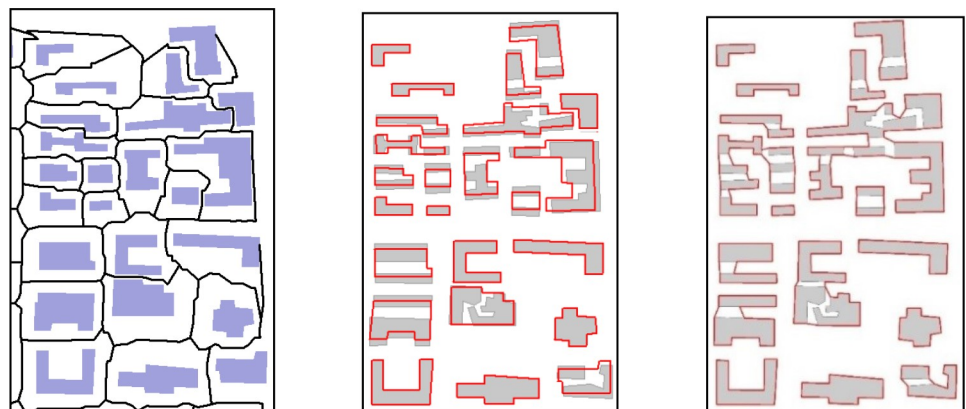

Fig 13. A comparison of building aggregation between the proposed method (B) and ArcGIS method (C).

Table 1. The quantitative comparison between our proposed method and that of ArcGIS.

Method	Original area (m²)	Aggregated area (m²)	Change rate
ArcGIS method	37596.6	44137.7	17.4%
Proposed method	37596.6	39476.3	5.1%

The building generalization is aimed at resolving the issue of crowded building distribution in a small space. If the room competition and spatial conflict is serious in the absence of additional room for displacement, the proposed method then does not work effectively. For such a situation, the conflict skeletons account for a large rate, and the building group detection is required to take into consideration a greater number of aspects. One improvement strategy is to use the graph structure to analyze the building pattern by conflict object connection. A dual geometric construction and Delaunay triangulation can be reached by connecting the center points within the tessellated polygon. Accordingly, a number of conjoint networks can be produced by connecting the geometric central points of the conflicting building on the basis of the building partitioning model, as is presented in Fig 14. With the combination of other approaches, the future assignment is to discover building distribution patterns based on the proposed model of neighborhood representation.

5. Conclusion

In spatial data handling, the neighborhood relationship plays an important role in map generalization, co-location detection, and other applications. In addition to the buffering operation, the triangulation is also an efficient tool in neighborhood analysis. Delaunay triangulation, as a
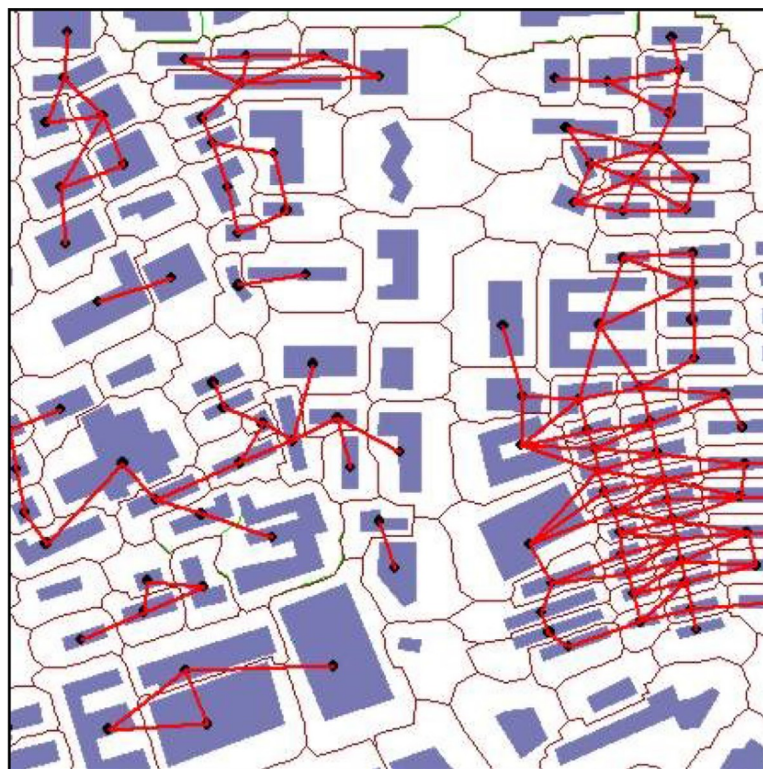

Fig 14. The network of a connective conflict building object.

special geometric construction, is widely applied in map generalization for detecting neighboring regions or conflict regions. A large number of concrete applications of Delaunay triangulation in generalization algorithm design and data model development have been explored. However, the previous application models have not been formally represented. From the computation perspective, it is necessary to establish a formal model based on the Delaunay triangulation for neighborhood representation.

In this study, we build an FTDM data model based on the representation of the spatial field on the fundamentals of Delaunay triangulation, which combines the basic concept of a regular grid raster model and the advantage of the Delaunay triangulation on the spatial neighbor relationship representation. This paper presents the formal definition and operating algorithm of three types of operators, which correspond with the raster operation of a regular raster. The three operators play two roles: one takes a triangular vertex, edge, and area as the components of spatial objects of point, line, and region; the other takes the connection between the triangles as the components of the spatial neighbor relationship between spatial objects. This model can be used widely in the map generalization, detection of conflict in spatial data mining, merging of non-connected objects, and the representation of different spatial neighbor relationship classes.

To illustrate the application of the FTDM model, we examine the generalization of urban building clusters, which proves to be reasonable and effective. In this proposed method, we consider the building generalization through three steps of operations, i.e., grouping, displacement, and aggregation. In the decision-making stage, the grouping is used to detect the cluster pattern, which has to be preserved after abstraction. In the geometric operation stage, the displacement operation attempts to move buildings within a group together to avoid increasing the built-up area, and the aggregation operation attempts to combine the closed building group into one block. The algorithm complexity of the proposed method is O(nlogn), which satisfies the needs of general map generalization. The proposed FTDM model and the formal operator *Expand(r)* and *Skeleton(r)* can provide for conducting the three aforementioned operations. Through the triangle tessellation in the FTDM model and the neighborhood analysis by *Expand(r)*, we can detect the building group and conflict area and further, through *Skeleton(r)*, move closed buildings together. Using the FTDM-based generalization operations, the consistency of the overall distribution pattern of buildings can be well maintained. The relative position relationships of buildings are well considered and the neighborhood topological relationships of buildings will not be destroyed. However, in this method, it is difficult to establish one to one relationship of objects before and after building generalization and quantitatively evaluate the scale relationship change of each object due to the integrated processing of displacement and aggregation. Through the experiment comparison between our method and that of the ArcGIS function, the built-up area increment by our method is less than one third of that in the ArcGIS method (5.1% VS 17.4%). In addition to areal features, the FTDM model can also be applied for map generalization of point, line features, such as line simplification [33, 34], building displacement [29] and point cluster simplification [44]. As this study focuses on the establishment of FTDM model and the space is limited, the detailed applications in map generalization of point, line features will not be described in this paper.

However, some problems are still encountered while applying the proposed data model, such as that the original distribution pattern of the urban building cluster may be damaged and the position accuracy of some early aggregated buildings cannot be maintained as they are displaced in the whole process [28, 45]. In the future, the proposed method should be improved with respect to the distribution pattern and positional accuracy, especially in the remaining distribution pattern. In addition, in the process of building generalization, many aspects such as the geometrical, semantic characteristics and distribution pattern of buildings

should be considered. A perfect generalization strategy based on the proposed data model should be further investigated.

Acknowledgments

Author Contributions

Writing – original draft: Tinghua Ai.

Writing – review & editing: Tinghua Ai, Hongmei Yin, Yilang Shen, Min Yang, Lu Wang.

References

1. Burrough PA. Principles of geographical information systems for land resources assessment. Oxford: Oxford University Press. 1986.

2. Regnauld N, Edwardes A, Barrault M. Strategies in building generalisation: modelling the sequence, constraining the choice. In Proceedings of the 19the ICA Workshop on Progress in Automated Map Generalization. Ottawa. 1999.

3. Lee D. New cartographic generalization tools, CD-Rom. In Proceedings of the 19th ICC Conference. Ottawa. 1999.

4. Guo R, Ai T. Simplification and aggregation of building polygons in automatic map generalization. J Wuhan Tech Univ Surv Mapp. 2000; 25(1): 25–30.

5. Buchin K, Meulemans W, Renssen A, Speckmann B. Area-preserving simplification and schematization of polygonal subdivisions. ACM Trans Spat Algorithms Syst. 2016; 2(1): 1–35.

6. Chen W, Long Y, Shen J, Li W. Structure recognition and progressive simplification of the concaves of building polygon based on constrained D-TIN. Geomat Inf Sci Wuhan Univ. 2011; 36(5); 584–587.

7. Xu W, Long Y, Zhou T, Chen L. Simplification of building polygon based on adjacent four-point method. Acta Geod Cartogr Sin. 2013; 42(6): 929–936.

8. Damen J, van Kreveld M, Spaan B. High quality building generalization by extending the morphological operators. In Proceedings of the 12th ICA Workshop on Generalisation and Multiple Representation. Montpellier, France. 20–21 June 2008.

9. Meijers M. Building simplification using offset curves obtained from the straight skeleton. In Proceedings of the 19th ICA workshop on Generalisation and Multiple Representation. Helsinki, Finland. 14 June 2016.

10. Kada M. Aggregation of 3D buildings using a hybrid data approach. Cartogr Geogr Inf Sci. 2011; 38(2): 153–160.

11. Wang H, Wu F, Zhang L, Deng H. The application of mathematical morphology and pattern recognition to building polygon simplification. Acta Geod Cartogr Sin. 2005; 34(3): 269–276.

12. Sester M. Optimization approaches for generalization and data abstraction. Int J Geogr Inf Sci. 2005; 19(8-9): 871–897.

13. Sester M. Generalization based on least squares adjustment. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing. Amsterdam, the Netherlands. 16–23 July 2000.

14. Bayer T. Automated building simplification using a recursive approach. In: Gartner G, Ortag F, editors. Cartography in Central and Eastern Europe. Springer: Berlin/Heidelberg, Germany; 2010. p. 121–146.

15. Liu P, Ai T, Deng J. Simplification and rectangularity of building-polygon based on least squares adjustment. J China Univ Min Technol. 2008; 37(5): 699–704.

16. Haunert JH, Wolff A. Optimal and topologically safe simplification of building footprints. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. San Jose, CA, USA. 3–5 November 2010.

17. Cheng B, Liu Q, Li X, Wang Y. Building simplification using backpropagation neural networks: A combination of cartographers' expertise and raster-based local perception. GISci Remote Sens. 2013; 50(5); 527–542.

18. Shen Y, Ai T, Li C. A simplification of urban buildings to preserve geometric properties using superpixel segmentation. Int J Appl Earth Obs Geoinformation. 2019; 79: 162–174.

19. Regnauld N. Recognition of building cluster for generalization. In Proceedings of the 7th International Symposium on Spatial Data Handling. Coimbra, Portugal. 1996.

20. Li Z, Yan H; Ai T, Chen J. Automated building generalization based on urban morphology and Gestalt theory. Int J Geogr Inf Sci. 2004; 18(5): 513–534.

21. Chaudhry O, Mackaness W. Visualisation of settlements over large changes in scale. In Proceedings of International Cartographic Association Generalization Workshop. La Coruña, Spain. 7-8 July 2005.

22. Shen Y, Ai T, Li W, Yang M, Feng Y. A polygon aggregation method with global feature preservation using superpixel segmentation. Comput Environ Urban Syst. 2019; 75: 117–131.

23. Ai T, Guo R. A constrained Delaunay partitioning of areal objects to support map generalization. J Wuhan Tech Univ Surv Mapp. 2000; 25(1):35–41.

24. Preparata FP, Shamos MI. Computational geometry. an introduction. Springer: New York, USA. 1985.

25. Jones CB, Bundy GL, Ware MJ. Map generalization with a triangulated data structure. Carto GIS. 1995; 22(4):317–331.

26. Ware JM, Jones CB. A spatial model for detecting (and resolving) conflict caused by scale reduction. In: Kraak MJ, Molenaar M. editors. Advance in GIS Research II (7th Int. Symposium on Spatial Data Handling). Taylor & Francis: London, England; 1997. p. 547–558.

27. Peng W. Automatic generalization in GIS. ITC Publication Series, Enschede. 1997.

28. Zhang X, Stoter J, Ai T, Kraak MJ, Molenaar M. Automated evaluation of building alignments in generalized maps. Int J Geogr Inf Sci. 2013; 27(8): 1550–1571.

29. Ai T, Zhang X, Zhou Q, Yang M. A vector field model to handle the displacement of multiple conflicts in building generalization. Int J Geogr Inf Sci. 2015; 29(8): 1310–1331.

30. Bader M, Barrault M, Weibel R. Building displacement over a ductile truss. Int J Geogr Inf Sci. 2005; 19 (8-9): 915–936.

31. Kovanen J, Sarjakoski LT. Sequential displacement and grouping of point symbols in a mobile context. J Loc Bas Serv. 2013; 7(2): 79–97.

32. Shen Y, Ai T, Wang L, Zhou J. A new approach to simplifying polygonal and linear features using superpixel segmentation. Int J Geogr Inf Sci. 2018; 32(10): 2023–2054.

33. Ai T, Ke S, Yang M, Li J. Envelope generation and simplification of polylines using Delaunay triangulation. Int J Geogr Inf Sci. 2017; 31(2): 297–319.

34. Ai T, Zhou Q, Zhang X, Huang Y, Zhou M. A simplification of ria coastline with geomorphologic characteristics reserved. Mar Geodesy. 2014; 37(2): 167–186.

35. Shen Y, Ai T, He Y. A new approach to line simplification based on image processing: A case study of water area boundaries. ISPRS Int J Geo-Inf. 2018; 7(2): 41.

36. Meijers M, Savino S, Van Oosterom P. SPLITAREA: an algorithm for weighted splitting of faces in the context of a planar partition. Int J Geogr Inf Sci. 2016; 30(8): 1522–1551

37. Ai T, Liu Y, Chen J. The hierarchical watershed partitioning and data simplification of river network. In: Riedl A, Kainz W, Elmes GA, editors. Progress in Spatial Data Handing. Springer: Berlin Heidelberg, Germany; 2006. p. 617–632.

38. Aichholzer O, Aurenhammer F, Alberts D, Gärtner B. A novel type of dkeleton for polygons. In: Maurer H, Calude C, Salomaa A, editors. J.UCS The Journal of Universal Computer Science. Springer, Berlin, Heidelberg; 1996.

39. Shen Y, Ai T, Yang M. Extracting centerlines from dual-Line roads using superpixel segmentation. IEEE Access. 2019; 7: 15967–15979.

40. Foerster T, Stoter J, Kraak MJ. Challenges for automated generalisation at European mapping agencies: a qualitative and quantitative analysis. Cartogr J. 2010; 47(1): 41–54.

41. Ai T, Zhang X. The Aggregation of Urban Building Clusters Based on the Skeleton Partitioning of Gap Space. In: The European Information Society: Leading the Way with Geo-Information. Springer: Berlin/Heidelberg, Germany; 2007. p. 153–170.

42. Yang M, Ai T, Yan X, Chen Y, Zhang X. A map-algebra-based method for automatic change detection and spatial data updating across multiple scales. Trans GIS. 2018; 22(2): 435–454.

43. Tong X, Jin Y, Li L, Ai T. Area-preservation simplification of polygonal boundaries by the use of the structured total least squares method with constraints. Trans. GIS. 2015; 19(5): 780–799.

44. Ai T, Liu Y. A method of point cluster simplification with spatial distribution properties preserved. Acta Geod Cartogr Sin. 2002; 31(2):175–181.

45. He Y, Ai T, Yu W, Zhang X. A linear tessellation model to identify spatial pattern in urban street networks. Int J Geogr Inf Sci. 2017; 31(8): 1541–1561.