

Article

A Global Continuous Multi-Scale Terrain Contour Simplification Method Based on the Level Set

Haoyue Qian ^{1,2}, Lin Yang ^{2,3,*}, Zejun Zuo ^{2,3}, Run Wang ^{2,3}, Wenjie Zhen ^{2,3} and Shunping Zhou ^{2,3}

- ¹ Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Land and Resources, 8007 Hongli West Road, Shenzhen 518034, China; qhy_2015@cug.edu.cn
- ² School of Geography and Information Engineering, China University of Geosciences, 388 Lumo Road, Wuhan 430074, China; zjzuo@cug.edu.cn (Z.Z.); wangrun@cug.edu.cn (R.W.); zhenwenjie@cug.edu.cn (W.Z.); zhoushunping@mapgis.com (S.Z.)
- ³ National Engineering Research Center of Geographic Information System, China University of Geosciences, 388 Lumo Road, Wuhan 430074, China
- * Correspondence: yanglin@cug.edu.cn; Tel.: +86-027-6788-3728

Abstract: Multi-scale spatial representation has been widely used in geographic information and online mapping systems. Terrain contour, which provides a reference for understanding and monitoring the Earth's surface, is an important data category. For the multi-scale representation of contour lines, simplification is a fundamental step in providing different levels of detail for linear features. However, achieving a global continuous multi-scale simplification of contours remains a challenge. Therefore, based on the concept of level set, a novel contour simplification method labeled the continuous changing surface model (CCSM) was proposed in this paper. The CCSM was built by using a non-uniform rational B-spline constrained with characteristics and was then intersected with a set of horizontal planes with progressive height values. The generated intersection lines are considered continuous multi-scale simplified contours. Experiments were conducted on a 1:50,000 real contour dataset to verify the effectiveness of CCSM. Results showed that the changes in the shape of the simplified contours generated by CCSM are more natural and progressive than those generated by two other significant simplification methods. CCSM can also effectively balance local and global structures and has potential applications in obtaining a continuous multi-scale representation of terrain contours.

Keywords: cartographic generalization; contour; continuous simplification; multi-scale; level set; NURBS



Citation: Qian, H.; Yang, L.; Zuo, Z.; Wang, R.; Zhen, W.; Zhou, S. A Global Continuous Multi-Scale Terrain Contour Simplification Method Based on the Level Set. *Appl. Sci.* **2021**, *11*, 3855. <https://doi.org/10.3390/app11093855>

Academic Editor: Edoardo Rotigliano

Received: 20 March 2021

Accepted: 21 April 2021

Published: 24 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Multi-scale spatial representation has a wide range of applications, such as zooming in/out, and is among the most common operators in geographic information and online mapping systems [1–4]. Rendering spatial data at different scales also improves our understanding and analysis of spatial data, thereby meeting the continuous visual needs of modern map users. As essential geomorphic elements, contour lines are among the most often used data for describing the geographic information of the Earth's surface [5–7]. A simplification of these contour lines provides different levels of detail for linear features, which is an effective operator for maintaining essential shapes in multi-scale terrain maps [1,8,9]. Although contour simplification methods have been investigated for a long time, studies on the continuous multi-scale representation of contour lines still face considerable limitations.

Various contour simplification methods have been proposed over the past few decades. These methods can mainly be divided into (1) reduction in point set [10–15], (2) bending simplification [8,16–20], and (3) scale-driven simplification [9,21–24].

The first strategy, point set reduction, is a straightforward approach for achieving simplified polylines. The classical Douglas–Peucker (DP) algorithm [10] introduced a

pre-defined threshold for selecting a set of significant vertices recursively for simplification. However, the simplified contours seem stiff and may contain self-intersection and topological conflicts when dealing with dense contours for multi-scale simplification. To improve the DP algorithm, Saalfeld et al. [12] utilized dynamic convex hull algorithms, neighborhood screening, and efficient measures of topological consistency between features to avoid potential topological conflicts. Wu et al. [14] employed convex hull computations by partitioning a polyline into a set of separable star-shaped sub-polylines and applying the DP algorithm for each sub-polyline to avoid self-intersection along the recursive refinements. Pallero et al. [15] proposed an enhanced simplification algorithm based on the DP algorithm to obtain robust results regardless of the morphology of the initial line. However, when simplifying dense contours, the above methods generate results with discontinuous deformation. Meanwhile, the essential shape features of simplified contours may shift or contain topological errors.

For the second strategy, bending simplification, Wang and Muller [18] proposed the Wang–Muller (WM) simplification algorithm to maintain the structure of simplified bends based on size, shape, and context. Zhu et al. [20] leveraged the furthest visibility principle to divide the bend area of the contour line and to maintain the main characteristics of simplified contours at the target scale without self-intersection. Ai et al. [8] presented a simplified algorithm based on Delaunay triangulation and Perkal's ε -circle rolling algorithm to preserve the essential bend of a polyline and maintain the area amid large-scale changes during the simplification process. However, these bend reduction methods for simplifying polylines focus on simplifying the local bend than the global structure continuity.

The third strategy, scale-driven simplification, has become popular in recent years. Li and Openshaw [21] proposed a “natural principle” that can be employed to the overall shape simplification of linear features to address the stiff simplified results of the DP algorithm. Bertolotto and Egenhofer [22] proposed a model that can be used to represent multi-scale maps and guarantee the consistency of generalized data in the simplification process. Nöllenburg et al. [25] focused on polyline morphing at different scales and presented a dynamic programming algorithm with the corresponding linear features. Deng et al. [23] proposed an improved morphing method for two linear features at different scales based on their entire structural information. Li et al. [9] used simulated-annealing-based morphing (SABM) to implement continuous linear features generalization. Du et al. [24] proposed a progressive simplification method based on multi-bend groups that can delete bend units as little as possible. This method has advantages in area preservation during the multi-scale representation. Liu et al. [1] proposed a Fourier-based method using head/tail breaks that simplifies polylines via a Fourier approximation according to the global shape. These scale-driven simplification methods have advantages over the two aforementioned strategies (i.e., point set reduction and bending simplification). However, they lack domain-specific constraints in the scale-driven simplification process, thereby producing simplified results that do not completely follow the essential global structure of the initial line at continuous scales.

In sum, the three aforementioned simplification strategies still show limitations in continuous multi-scale representation. On the one hand, simplifications occur only through a few fixed scales. This problem makes it impossible to show data from one scale to another smoothly and ignores the actual morphological transformations of the contour line of desired scales between fixed scales. On the other hand, the local structures are overemphasized, and the global structures of linear characteristics are ignored during the simplification of continuous multiple scales, which may lead to a sudden change in the shape of the contour line ('jumps') during transformation. Given these limitations, this study proposes a novel contour simplification method to retain the global structure of linear characteristics and to implement a multi-scale deformation of contours continuously. This method is labeled the continuous changing surface model (CCSM) based on the notion of the level set method.

The core idea of the level set method [26,27] is that the motion of lower-dimensional (e.g., 2D) curves can be derived from a higher dimensional (e.g., 3D) surface. Accordingly, our simplification model derives 2D simplified contours from a meticulously constructed 3D surface labeled the continuous changing surface (CCS). According to the contour features, a CCS is established between two contours and is then intersected with a set of horizontal planes with continuous height values. The generated intersection lines are considered simplified contours with continuous scales. We also compare our proposed method with two traditional significant polyline simplification methods on a real contour dataset in China to validate the effectiveness of CCSM. Results show that the shape changes of contours simplified by CCSM are natural and continuous between scales and that the CCSM maintains the global shape features much better than the other two classical significant simplification algorithms.

The major original contributions of this study are twofold. First, we propose the CCSM simplification method guided by the level set framework to derive simplified contours with continuous multi-scale transformation, thereby reducing the loss of deformed details during the continuous multi-scale simplification process. Second, we propose a new contour deformation analysis method, namely, a shape similarity-scale trend line based on the improved Fréchet distance, which provides a tool for evaluating the performance of multi-scale polyline simplification methods.

The rest of this paper is organized as follows. Section 2 elaborates on the proposed method. Section 3 describes the experiment using the contour dataset and reports the results. Section 4 discusses two key factors affecting the quality of CCSM, the arbitrary scale simplification, and the retention of topographical features. Section 5 concludes the paper.

2. Methodology

The CCSM methodological framework was proposed for global continuous multi-scale contour simplification. The main idea of this framework is to derive 2D contours from a 3D surface by using the level set method. The 3D surface was constructed by the non-uniform rational B-spline (NURBS), a common mathematical form for representing standard analytical shapes and free-form curves/surfaces [28,29]. The built 3D surface, namely, the continuous changing surface (CCS), was used to guide the entire simplification process. The framework consists of the following main steps as shown in Figure 1:

(1) Extracting characteristics. The extraction of contour characteristics aims to preserve the global morphology features of simplified contour lines. These characteristics include characteristic points, characteristic sub-polylines, and characteristic links. To define CCSM constraints, the most significant morphology feature points in the shape were extracted from the initial contour to form a target contour by using the DP algorithm. The initial and target contours were converted into NURBS curves, and the two contour lines were assigned two height values. Accordingly, the NURBS curves were segmented into characteristic sub-polylines based on the characteristic points. Characteristic links were then constructed by linking the characteristic points of these two curves in order.

(2) Building the CCS. The extracted characteristics were used to weave a wireframe (WF) that constrains the construction of the CCS. Faced with this constraint, the CCS was built by using the NURBS function.

(3) Generating global continuous multi-scale simplified contours. First, the CCS was converted into a triangle mesh (CCS-mesh), which was then intersected with a set of horizontal planes with continuous height values. The generated intersection lines served as global continuous multi-scale simplified contours.

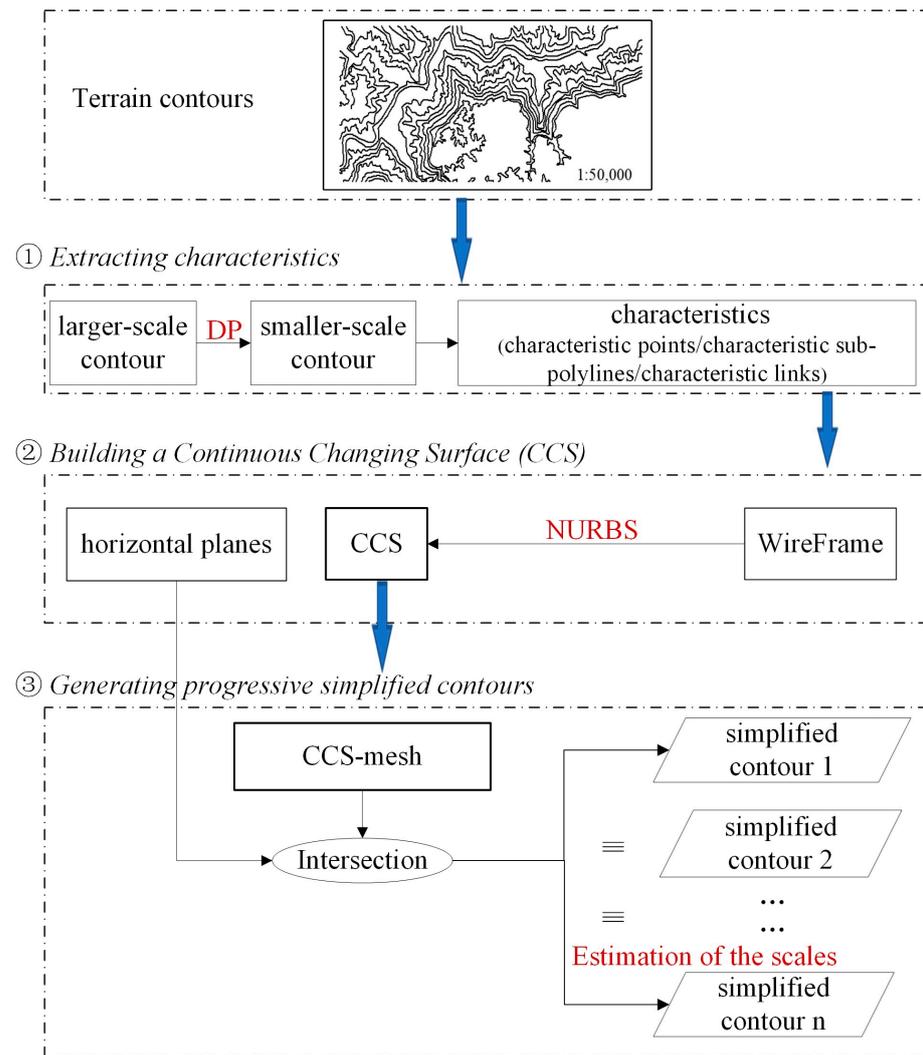


Figure 1. CCSM methodological framework.

2.1. Extracting Characteristics

Linear features are considered basic elements in maps and suitable features that need to be retained during multi-scale transformation [23]. Therefore, when constructing the CCS, the optimum extraction of morphology characteristics and the constraint of characteristics are critical to obtain the final global continuous multi-scale simplified results. The characteristics of CCS include characteristic points, characteristic sub-polylines, and characteristic links.

(1) Characteristic points. The boundary of the CCS needs to be constructed with the initial contour L_0 and the target contour L_n . L_0 represents the initial contour with a large scale, whereas L_n represents the target contour with a small scale. These two contours served as boundaries for building the CCS.

(2) Characteristic sub-polylines. L_0 and L_n were resampled via equidistant sampling to ensure that they have the same number of points. To maintain the morphology shape of the contour line, the number of resampling points should exceed the number of characteristic points extracted from the initial contour. All resampling points were considered characteristic points. Then, two NURBS curves (L'_0 and L'_n) were defined using the characteristic points from L_0 and L_n , respectively. It is worth noting that L_0 and L'_0 are highly similar in shape and size, as are L_n and L'_n . Meanwhile, the B-spline curve was built based on the control points $P_i (i = 0, 1, \dots, m)$, which are the characteristic points of L_0 or L_n , where m

represents the number of characteristic points of L_0 or L_n . Generally, the NURBS curve ($p(u)$) is defined over the knot $U = \{u_0, u_1, \dots, u_m\}$ vector as Equation (1).

$$p(u) = \sum_{i=0}^m P_i N_{i,k}(u) \tag{1}$$

where $N_{i,k}(u)$ is the B-spline base function of order k (where k is set to 3), which is represented as Equation (2).

$$\begin{cases} N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases} \\ N_{i,k}(u) = \frac{(u-u_i)N_{i,k-1}(u)}{u_{i+k}-u_i} + \frac{(u_{i+k+1}-u)N_{i+1,k-1}(u)}{u_{i+k+1}-u_{i+1}} \quad (k > 1) \end{cases} \tag{2}$$

Accordingly, the characteristic sub-polylines of L'_0 and L'_n were obtained based on these characteristic points.

(3) Characteristic links. The direct lines in a sequence linked the characteristic points. These links represent the correspondence relationship between the characteristic points of L'_0 and L'_n , and are thereby labeled characteristic links.

In sum, as shown in Figure 2, the characteristic points, characteristic sub-polylines, and characteristic links were used as feature constraints that are inherited from the initial and target contours to constrain the global continuous multi-scale simplification process. To build the CCS in the 3D space, Z_{min} and Z_{max} were assigned the bottom and top height values of CCS. Z_{min} and Z_{max} correspond to the height values of L'_0 and L'_n , respectively. Accordingly, the characteristic points, characteristic sub-polylines, and characteristic links on L'_0 and L'_n also inherit the height values Z_{min} and Z_{max} .

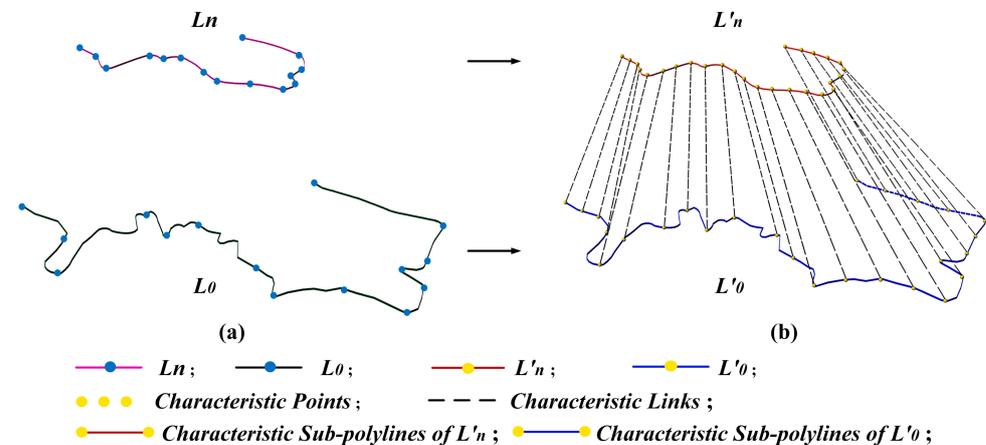


Figure 2. Characteristics between the initial contour $L_0(L'_0)$ and target contour $L_n(L'_n)$. (a) Initial contour L_0 and target contour L_n . (b) Characteristics between the initial contour L'_0 and target contour L'_n .

2.2. Building The CCS

2.2.1. Level Set Method

Osher and Sethian proposed the level set method [26], whose main idea is to raise some computations in lower dimensions to a higher one and consider the N-dimensional description as a level of N+1 dimensions, which is an effective implicit representation of curves and surfaces [26,27]. In other words, the motion of lower-dimensional (e.g., 2D) curves can be derived from a higher-dimensional (e.g., 3D) surface, thereby allowing the derivation of 2D simplified contours from the 3D surface. The implicit function is essential in guiding the motion of curves on the surface. Based on the notion of the level set method, the NURBS function was selected in this study as an implicit function to generate CCS.

2.2.2. Building the CCS Based on Characteristics

To constrain the shape change of the contour during the simplification process, the CCS was built based on characteristics. The CCS building process was divided into three steps.

Step 1. Setting characteristics in the 3D space. As described in Section 2.1, all the characteristic points, characteristic sub-polylines, and characteristic links on L'_0 and L'_n inherit the height values from Z_{min} and Z_{max} .

Step 2. Building a wireframe (WF) from 3D characteristics. As shown in Figure 3, four points were sampled on each characteristic link with equal intervals to construct a third-order NURBS surface uniformly. These points were merged into the constraints of characteristics, which act together during the construction of the WF.

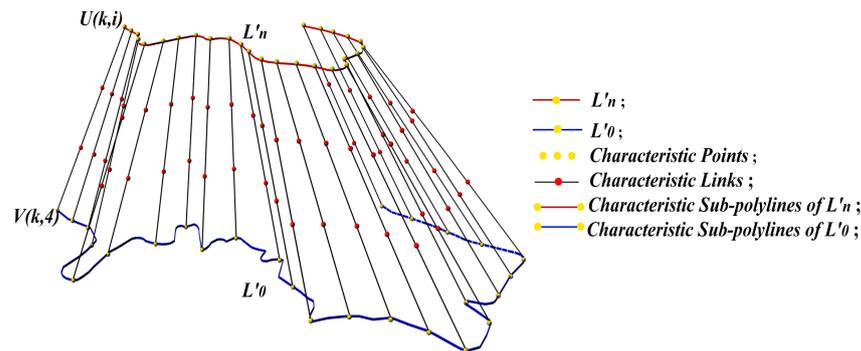


Figure 3. WF comprising the characteristic points, characteristic sub-polylines, and characteristic links.

Step 3. Constructing the CCS based on the WF. By using the NURBS function, the CCS was constructed in consideration of the WF constraints as shown in Equation (3).

$$CCS = NURBS_{function}(WF) \tag{3}$$

where $NURBS_{function}(\cdot)$ indicates the operator of acquiring the NURBS surface by the constraints of WF, which is defined as Equation (4).

$$P(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m \omega_{ij} P_{ij} N_{ik}(u) N_{jl}(v)}{\sum_{i=0}^n \sum_{j=0}^m \omega_{ij} N_{ik}(u) N_{jl}(v)} \tag{4}$$

where $P_{ij}(0 \leq i \leq n, 0 \leq j \leq m)$ represent the points on WF, n represents the number of characteristic points of L'_0 , which is the same as that of L'_n , m represents the number of points on characteristic links and m is set to 4, ω_{ij} represents the corresponding weight of P_{ij} , and $N_{ik}(u)$ and $N_{jl}(v)$ are the normalized B-spline base functions of orders k and l , respectively (both k and l are set to 3, $u_{k-1} \leq u \leq u_{n+1}$, $v_{l-1} \leq v \leq v_{m+1}$) and are defined over knot vectors $U = \{u_0, u_1, \dots, u_n, \dots, u_{n+k}\}$ and $V = \{v_0, v_1, \dots, v_m, \dots, v_{m+k}\}$. Based on these knot vectors, the following matrix $M_{n \times m}$ was obtained. As shown in Equation (5).

$$M_{n \times m} = \begin{vmatrix} u_1 v_1 & u_1 v_2 & \cdots & u_1 v_m \\ u_2 v_1 & u_2 v_2 & \cdots & u_2 v_m \\ \vdots & \vdots & \ddots & \vdots \\ u_n v_1 & u_n v_2 & \cdots & u_n v_m \end{vmatrix} = [u_i v_j] \tag{5}$$

As shown in Figure 4, a NURBS surface is constructed, which is referred to as the CCS in this study. The CCS plays a vital role in guiding the global continuous contour simplification process.

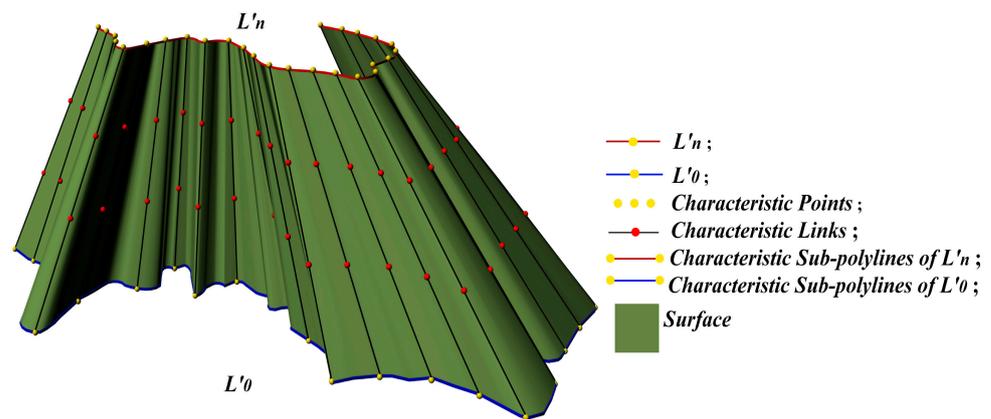


Figure 4. Established CCS.

2.3. Generating Global Continuous Multi-Scale Simplified Contours

2.3.1. Generation of Simplified Contours

Contour lines are essential parts of a terrain model. In terrain modeling, multiple levels of contours can be modeled on the 3D surface, and each contour depicts a terraced representation of the 3D surface at an elevation level. To obtain simplified contours, the CCS was intersected with a set of horizontal planes with successive height values to generate intersection lines. These intersection lines were taken as global continuous multi-scale simplified contours with varying simplification degrees of the initial contour.

To calculate the intersection lines between CCS and a set of horizontal planes, the CCS was tessellated into a polygonal mesh (e.g., triangle or quadrangle), that is, the $[u_i, v_j]$, values of NURBS were mapped to the corresponding grid points of the polygonal mesh. The triangle mesh was used to generate a polygonal mesh due to its advantages in the rapid generation and its intersection with planes. As shown in Figure 5, the triangle mesh comprises a set of triangular faces covering CCS and was therefore labeled CCS-mesh. This mesh may become irregular due to the uneven distribution of unconstrained sample grid points. To ensure the stability and convergence of the numerical solution, a suitable mesh generation technique should be applied to construct the appropriate surface mesh and to maintain a well-defined mesh that fits well in the CCS throughout the intersection process.

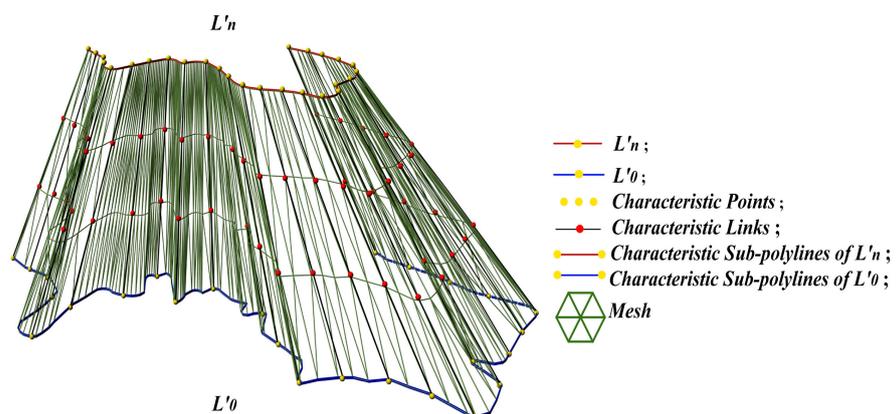


Figure 5. CCS-mesh covered by triangular faces.

Based on the CCS-mesh, we built a set of equidistant horizontal planes to intersect the CCS-mesh and the intersection lines that are regarded as simplified intermediate results. As defined in Equation (6) below, the result of every horizontal plane intersecting the CCS-mesh is an intersection line $L(Z_i)$. The height Z_i corresponds to each horizontal plane and is between the elevations of Z_{min} and Z_{max} . An intersection contour cluster was obtained by iterating the intersection between CCS-mesh and every horizontal plane. These intersecting

lines were considered simplified contour lines. The scale corresponding to these simplified contour lines continuously varied between the scales of the initial and target contours.

$$\text{Contour cluster} = \left\{ L_{(Z_i)} \mid L_{(Z_i)} = \text{CCS}_{\text{mesh}} \cap \text{Cutting plane } (Z_i) \right. \\ \left. Z_i = Z_{\min} + i * \text{interval}, i \leq \text{round}(Z_{\max} - Z_{\min}) / \text{interval} \right\} \quad (6)$$

Figure 6 presents an example of the simplified intermediate results generated by the CCS-mesh intersecting with a set of horizontal planes. Figure 6a represents a set of equidistant horizontal planes with continuous elevation; Figure 6b shows the intersection between the CCS-mesh and planes; Figure 6c illustrates the global continuous multiscale transformation of intersection lines. Each intersection line between the CCS-mesh and a horizontal plane captured the morphology characteristics of contours at a certain height level. Therefore, this cluster of simplified contours preserves the morphology characteristics corresponding to the continuously changing scales between the initial and target contours. The location or number of horizontal planes can be determined based on the desired scale simplified results or the number of simplified contours demanded by users.

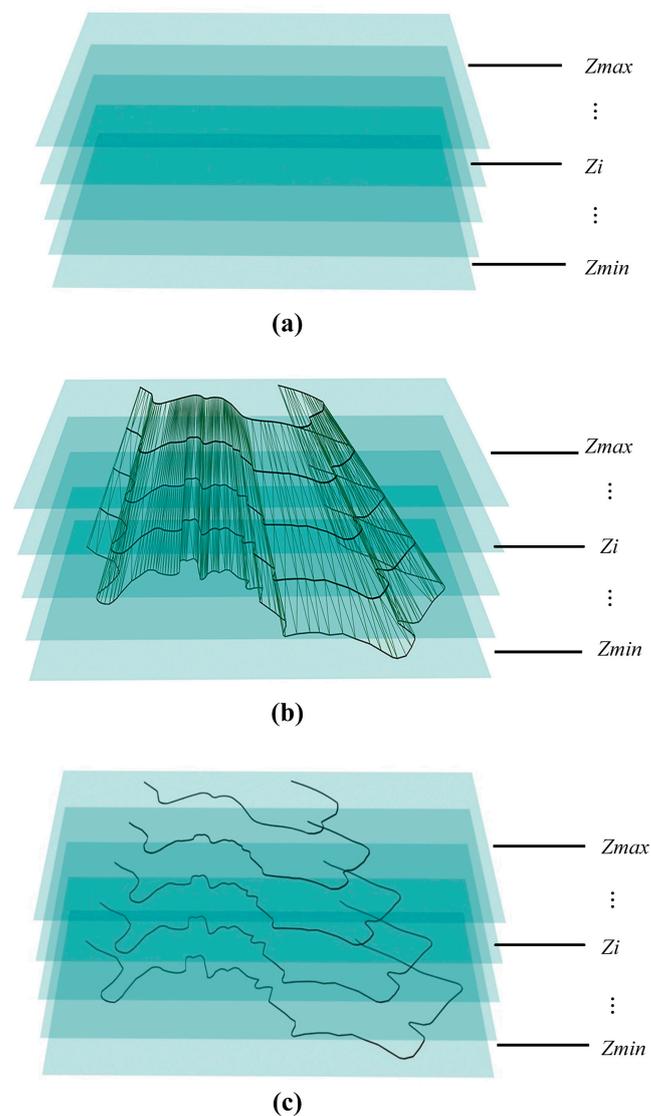


Figure 6. Intersection between the CCS-mesh and horizontal planes. (a) A set of horizontal planes. (b) The intersection between CCS-mesh and horizontal planes. (c) The simplified contours on horizontal planes.

2.3.2. Estimation of the Scales for the Simplified Contours

As a general rule, in cartographic generalization, the shape similarity of a contour should correspond to its scale value. The contour lines with complex shapes have a large scale, whereas those lines with simple shapes have a small scale. When the topographic map was simplified from large to small scale, the shape of contour lines on the topographic map also changed from complex to simple [30]. According to Li [31], similarity is a measure that has been widely used to analyze the shape relationship of topographical objects. In this subsection, the shape similarity between the simplified and initial contours was used as an intermediary to estimate the scale of an arbitrary simplified contour.

The shape similarity between two curves is often defined based on a specific distance measure [32]. One common metric of curve similarity is the Fréchet distance, which takes the flow of two curves into account and the pairs of points whose distance contributes to the Fréchet distance sweeping continuously along the respective curves [33]. However, the uneven distribution of the points of a polyline may affect the accuracy of the distance of this polyline [34]. To deal with such an uneven distribution, we proposed an improved Fréchet distance method based on equidistant resampling points to precisely reflect the shape similarity between two contour lines [35].

First, $L_{(Z_i)}$ represents the simplified contour corresponding to the horizontal plane with a height value of Z_i . The points on the simplified contour $L_{(Z_i)}$ were resampled via equidistant sampling, and the number of resampling points was similar to that in L'_0 . The initial contour line is $L'_0 : [0, n] \rightarrow V$, where V is a metric space and n is a positive integer. The sequence $(L'_0(0), L'_0(1), \dots, L'_0(n))$ of the characteristic points of the characteristic sub-polylines of L'_0 was denoted by $\delta(L'_0)$. $L_{(Z_i)}$ was defined similar to L'_0 . Let $\delta(L'_0) = (u_1, \dots, u_p)$ and $\delta(L_{Z_i}) = (v_1, \dots, v_q)$ be the corresponding sequences. A coupling L between L'_0 and $L_{(Z_i)}$ represents a sequence $(u_{a1}, v_{b1}), (u_{a2}, v_{b2}), \dots, (u_{am}, v_{bm})$ of distinct pairs from $\delta(L'_0) * \delta(L_{(Z_i)})$, where p and q denote the number of characteristic sub-polylines on the two contour lines. Therefore, the coupling should follow the order of points in L'_0 and $L_{(Z_i)}$. The number of points here refers to the number of characteristic points. The length $\|L\|$ of coupling L represents the length of the longest link in L as shown in Equation (7).

$$\|L\| = \max_{i=1, \dots, m} d(u_{ai}, v_{bi}) \tag{7}$$

Given two contour lines L'_0 and $L_{(Z_i)}$, their improved Fréchet distance is defined as Equation (8). This distance index can provide a basis for accurately capturing the similarity of the two contour lines.

$$\delta_{dF}(L'_0, L_{Z_i}) = \min \left\{ \|L\| \mid L \text{ is a coupling between } L'_0 \text{ and } L_{(Z_i)} \right\} \tag{8}$$

Then, the similarity between $L_{(Z_i)}$ and L'_0 is defined as Equation (9) based on the improved Fréchet distance.

$$Sim(L'_0, L_{Z_i}) = 1 - \delta_{dF}(L'_0, L_{Z_i}) / Length(L'_0) \tag{9}$$

where $\delta_{dF}(L'_0, L_{Z_i})$ represents the improved Fréchet distance between contours L'_0 and L_{Z_i} , and $Length(L'_0)$ represents the length of L'_0 . Note that $Sim(L'_0, L'_0) = 1$ and $Sim(L_{(Z_i)}, L'_0) = Sim(L'_0, L_{(Z_i)})$.

To reveal the correspondence between shape similarity and scale, we used an exponential regression function to fit the ideal relationship between the similarity of contours and scale [36,37]. Ideally, as the scales changed continuously from large to small, the shapes of the contour lines were transformed from complex to simple, and the similarity value gradually changed from large to small. As shown in Figure 7, the relationship between the contour scales and similarity values was fitted with an exponential regression function. The horizontal axis represents the scale, whereas the vertical axis represents the similarity value between the initial large-scale contour lines and the other small scales. As shown in

Figure 7, the trend of the relationship between the similarity values and scale is continuous and monotonic in the ideal case.

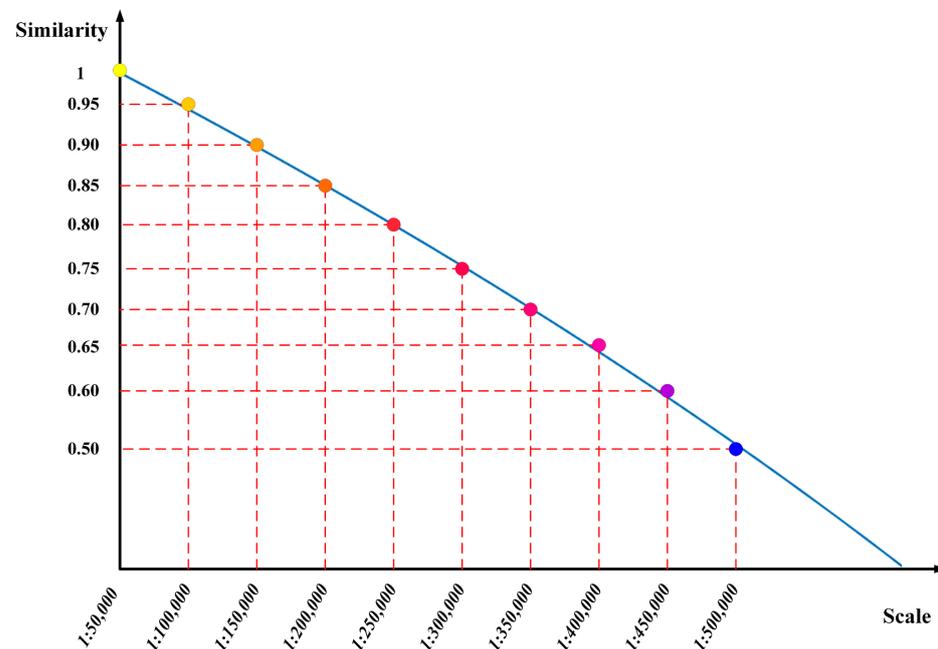


Figure 7. Ideal relationship between scale and similarity for continuous multi-scale simplification.

The transitions of gradual simplification from L'_0 to L'_n were emulated by the intersections of the CCS-mesh and the horizontal plane. To generate a simplified contour cluster from L'_0 to L'_n , each horizontal plane was assigned the height value Z_i in the equal interval sequence. Given that the relationship between the scale and similarity of different scale contours can be fitted with an exponential regression function and that the trend is continuously monotonic, according to Equation (9) and based on the similarities derived by Z_i values, an interpolation function $R_s(L'_0, L'_n, t)$ was formulated to characterize the transition. t is a scaling parameter with values ranging from 0 to 1 that can be used to bridge the continuous scales between L'_0 and L'_n by mapping Z_i . By referring to the scales of L'_0 and L'_n , t can be linearly converted into common actual scales. When t is 0, the scale is the initial scale of L'_0 , but when t is 1, the scale is the target scale of L'_n . In turn, the contour cluster can be ordered by t . According to the interpolation function R_s , the arbitrary scale of the simplified contours can be determined by the t value and its corresponding Z_i value. In sum, each simplified contour has a determined scale and similarity with L'_0 . Therefore, the proposed method can achieve arbitrary scale simplification.

3. Experiments and Results

3.1. Experimental Settings and Data

The CCSM was implemented by using the Open Cascade library given that its data structure is compatible with ISO 10303-42 guidelines. ISO 10,303 is known as the Standard for the Exchange of Product Model Data as published by the International Organization for Standardization (ISO) and the Initial Graphics Exchange Specification [38]. This standard can guarantee a certain level of generality of our proposed method. The experiment was conducted on a computer equipped with an Intel Core i5-6300U 2.40 GHz processor. As shown in Figure 8, the 1:50,000 topographic map of the northeastern part of Huimin, Chongqing, China, was downloaded from Google Earth and was used as our experimental contour dataset. Google Earth uses the Shuttle Radar Topography Mission as its digital elevation model data, which are mainly measured by NASA and the National Imagery and Mapping Agency. The digital elevation model data in China has an accuracy of 90 m [39].

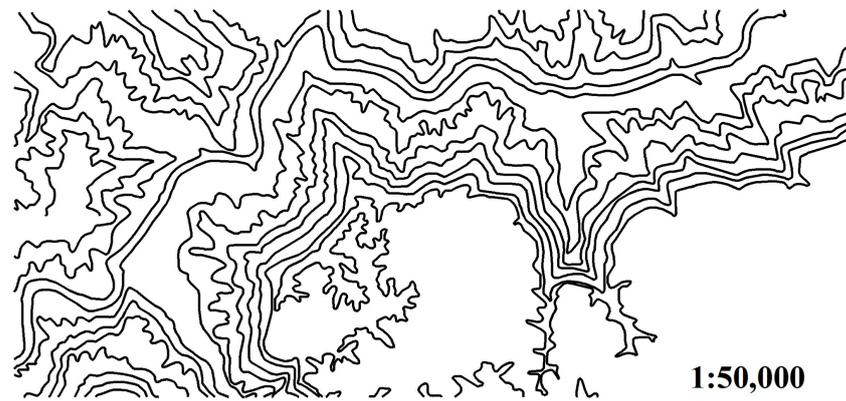


Figure 8. Contour line dataset with a scale of 1:50,000 in the northeastern part of Huimin, Chongqing, China.

3.2. Results and Analysis

To evaluate the effectiveness of our proposed method, the simplified contours generated by the Douglas–Peucker (DP) algorithm, Wang–Muller (WM) algorithm, and CCSM were compared. First, a visual analysis was performed to examine and verify the ability of these methods in showing the continuous multi-scale details of simplified contours. Second, the shape similarity change trend of multi-scale simplified contours generated by these methods was evaluated. Third, the retention degree of the global structural features of simplified contours generated by these methods was compared. Fourth, the change degree in position was assessed.

3.2.1. Results and Visual Analysis

This section compares the deformation process of the multi-scale simplified contours produced by the aforementioned three methods under the same source and target scale conditions. In this way, we guarantee that the simplified result generated by these methods is a transformation between the same two scales. The source contour scale was taken as the initial scale (i.e., 1:50,000), whereas the target contour scale was taken as the target scale (i.e., 1:200,000). The target contour of the CCSM was simplified by the DP algorithm and was the same as that of the DP algorithm. To easily detect changes in the shape of the simplified contours, the three simplification methods all produced 100 simplified contours (including the initial and target contours). Table 1 presents the parameter settings for generating these contours. For CCSM, the number of horizontal planes was set to 100. The DP algorithm was iterated 100 times, and the reduction of the reservation point decreased at a rate of 0.01. The WM algorithm was iterated 100 times, and the bend reduction rate was 0.01.

Table 1. Experimental parameter settings of three simplification methods for generating 100 simplified contours.

Parameter	Value
Number of horizontal planes (CCSM method)	100
Reduction rate of points (DP algorithm)	1%
Reduction rate of bends (WM algorithm)	1%

Figure 9 shows the experimental results obtained by the DP algorithm, WM algorithm, and CCSM. No. 1 denotes the initial contour and No. 100 denotes the target contour. The intermediate simplified contours between the initial and target contours are represented by serial numbers (Nos. 2 to 99). We only show here the key shape of a few intermediate simplified results obtained by the three methods during the entire simplification process. Figure 9a shows the simplified results obtained by the DP algorithm. According to Table 1, when the DP algorithm was used for simplification, the number of points gradually

decreased; ideally, the shape of the contour line should be changed accordingly. However, the shapes of the contours were almost unchanged until No. 81. Afterward, as indicated by the blue dotted rectangle, the shapes of the simplified contours changed starting from No. 85. Meanwhile, the shapes of the simplified contour line behind (Nos. 86 to 100) changed in a jump way. Figure 9b shows the simplified results obtained by the WM algorithm. According to Table 1, the bend number continuously decreased when the WM algorithm was used for simplification. The shapes continuously changed from Nos. 1 to 24. However, from Nos. 24 to 44, the shapes barely changed. From Nos. 44 to 45, the shape of the contour abruptly changed, and some key features were lost. Figure 9a,b indicate that the simplified contours generated by the DP and WM algorithms are discontinuous. Meanwhile, Figure 9c shows the simplified results obtained by the CCSM method. The ordinal number in this figure corresponds to the i value of Z_i , that is, the height of the intersecting horizontal planes (described in Section 2.3.1). Unlike the findings of the DP and WM algorithms, Figure 9c shows that the simplified contours generated by CCSM are progressive and include a level of detail that varies continuously. Moreover, the local characteristics were well preserved during the transformation through a few scale ranges (Nos. 1 to 70). Above all, the shapes of the simplified contours generated by CCSM continuously changed between the initial and target contours. CCSM can also yield better continuous multi-scale simplified contours compared with the DP and WM algorithms.

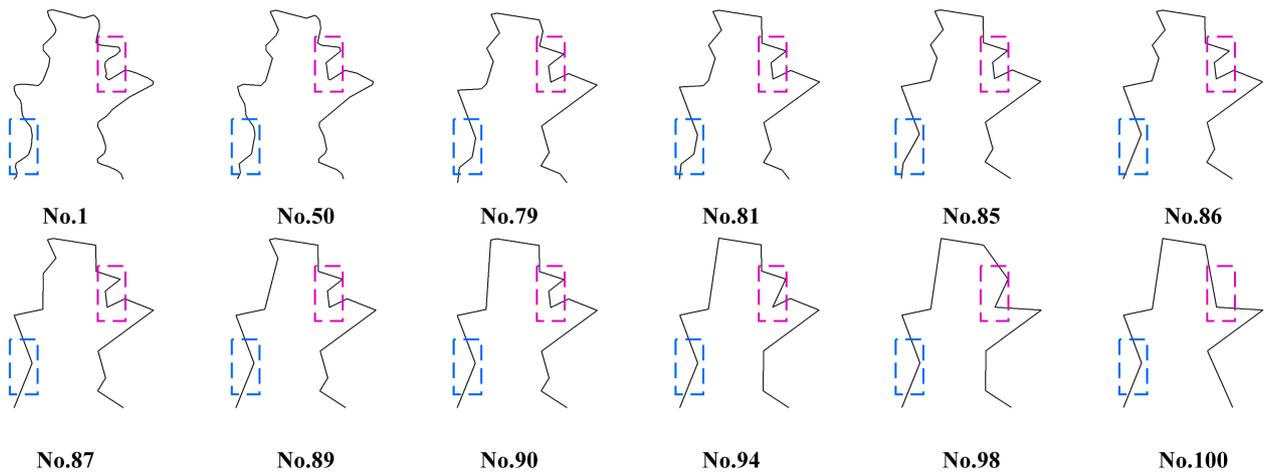
3.2.2. Comparison of Shape Similarity Change Trend

In this subsection, the shape similarity-scale trend lines obtained by the three methods were compared. The shape of an object is an important visual feature, and the difference in shape similarity can be used to describe shape change [40,41]. To avoid contingency, all initial contours in the study area were simplified by using the above three methods. All these methods generated 100 simplified results for each initial contour. The intermediate simplified contours between the initial and target contours were also represented by a serial number. The initial contour scale was taken as the source scale (i.e., 1:50,000), and the target contour scale was taken as the target scale. To further analyze the deformation of the simplified results, the target contour was the most simplified contour line containing two points (i.e., the starting and ending points of the initial contour), whereas the target contour scale was designated as a smaller scale (e.g., 1:20,000,000) or even approaching the infinitesimal scale (1:∞). Equation (10) was used to calculate the similarity between the simplified contour and the corresponding initial contour.

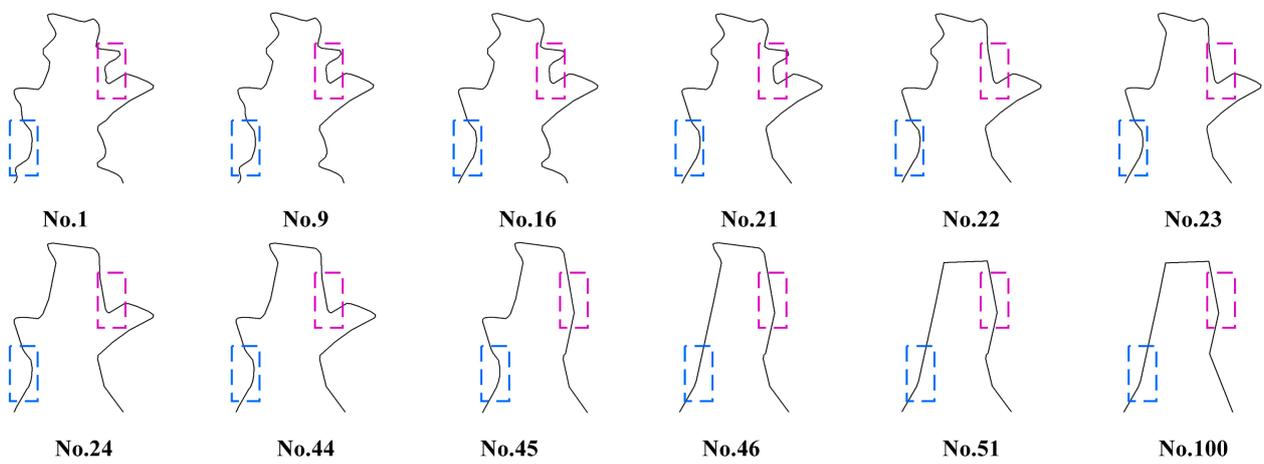
$$Sim(L_{initial}, L_i) = 1 - \delta_{dF}(L_{initial}, L_i) / Length(L_{initial}) \quad (i = 1, 2, \dots, 100) \quad (10)$$

where $L_{initial}$ represents the initial contour, L_i represents the simplified contour, and i represents the corresponding sequence number of the simplified contour.

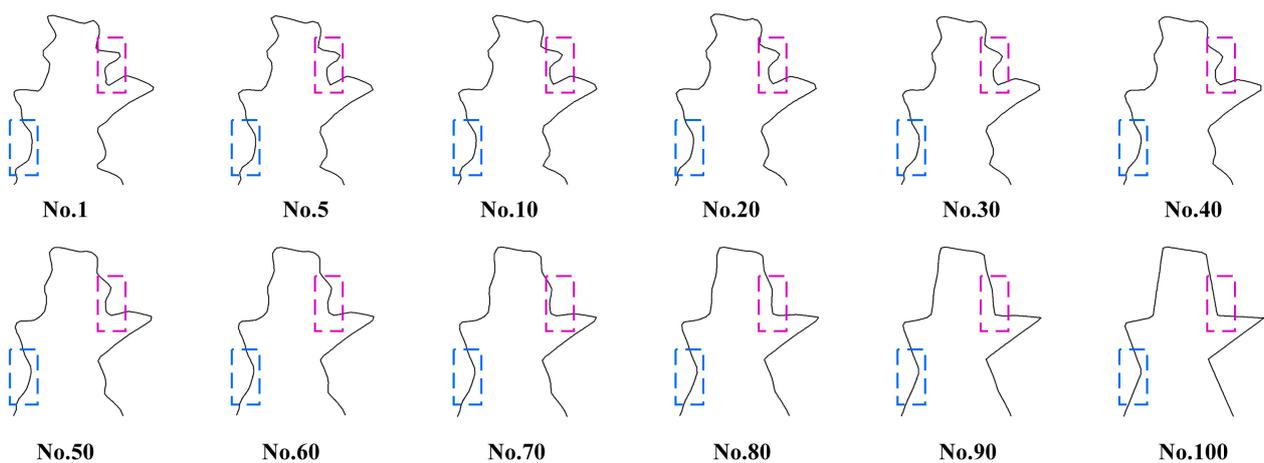
Figure 10 shows the shape similarity-scale trend line corresponding to the three simplification methods. The horizontal axis represents the sequence number ($i = 1, 2, \dots, 100$). The horizontal axis can also represent the scale because our setting parameters are continuously changing, and so should be the scale. The vertical axis represents the average shape similarity value of all initial contours at the corresponding sequence number i . When $i = 1$, the scale of the contour line is: 1:50,000; When $i = 100$, the scale of the contour line is: 1:20,000,000 or even approaching the infinitesimal scale (1:∞).



(a) The simplified contours generated by DP algorithm



(b) The simplified contours generated by WM algorithm



(c) The simplified contours generated by CCSM method

Figure 9. Simplified contours generated by the DP algorithm, the WM algorithm, and the CCSM method. (a) The simplified contours generated by DP algorithm. (b) The simplified contours generated by WM algorithm. (c) The simplified contours generated by CCSM method.

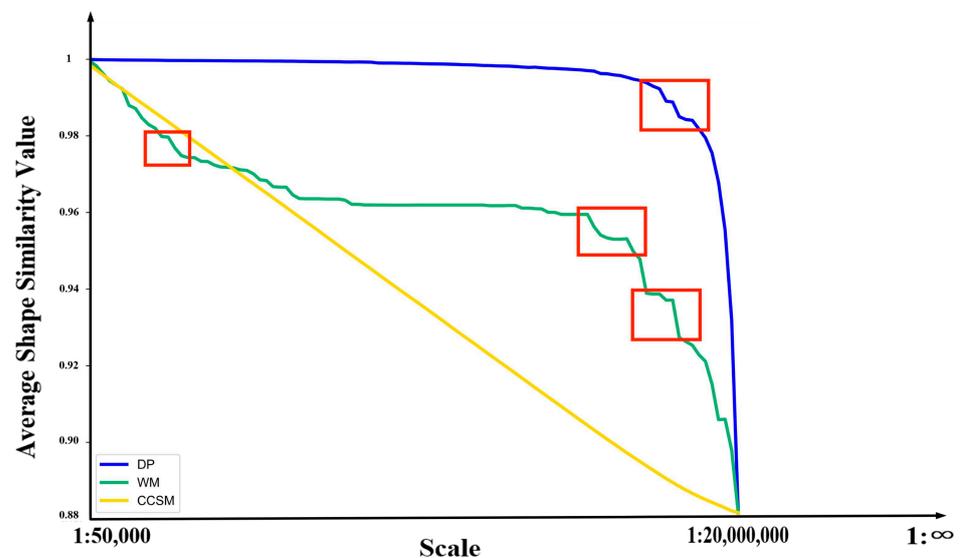


Figure 10. Average shape similarity trend of three simplification methods.

Overall, the average shape similarity value decreased along with the scale for all three simplification methods. As shown in Figure 10, both the blue and green curves jumped (denoted by red rectangles), which means that the shapes of the simplified contour lines generated by the DP and WM algorithms were not continuously changing. In other words, as the scale gradually changed, the changes in the shape of the simplified contours generated by the DP and WM algorithms were not continuous but abrupt, thereby leading to the loss of some intermediate scale details. In this case, the user demand for visual continuity is left unsatisfied. By contrast, the yellow curve in Figure 10 representing CCSM shows a monotonic and continuous trend that closely reflects the ideal relationship between similarity and scale at a continuous multi-scale (Figure 7). The average shape similarity gradually decreased along with the scale. Therefore, CCSM can obtain smooth and continuous multi-scale simplified contours.

3.2.3. Comparison of Global Structural Feature Retention

We then compared the ability of retaining the global structural features of the simplified contours generated by the DP algorithm, WM algorithm, and CCSM. To ensure a fair comparison, the retention degree of the global structural features on the generated simplified contours was examined at the same scale level. When simplifying line elements in electronic maps, the product of the minimum visual discriminant distance (usually set to 0.1 mm) and the reciprocal of the scale were utilized as simplification parameters. If the distance between two linear elements is less than this parameter, then their scale is the same [42]. In this study, the improved Fréchet distance between the simplified and initial contours was used as a proxy for the scale parameter. For example, when the initial contour scale was 1:50,000, the scale parameter was set to 5 m ($0.1 \text{ mm} \times 50,000 = 5 \text{ m}$). When the improved Fréchet distance between two contour lines was less than 5 m, the two scales can be considered the same. According to Equation (10), when calculating the similarity between an initial contour and an intermediate simplified result whose Fréchet distance was within 5 m, the minimum similarity value was approximately equal to 0.998, that is, the difference $\Delta s \approx 0.002 \cdot (1 - 0.998)$. Therefore, 0.002 was considered the maximum similarity value for identifying whether two simplified contours are at the same scale.

Three simplification methods were used to simplify 20 contour lines that were randomly selected from the dataset (as shown in the first column of Table 2). For each initial contour line, 100 simplified contours were obtained by each method. The scale of the initial contour scale was 1:50,000. The target contour was the most simplified contour line containing two points, namely, the starting and ending points of the initial contour. The similarities between all simplified contours and their initial contours were calculated by

using Equation (10). To avoid contingency in the experimental process, for each initial contour line, multiple groups of simplified contours were selected within the maximum similarity value (as shown in the second column of Table 2), and each group had three simplified contours generated by the three simplification methods. Therefore, each group had three similar values corresponding to the three methods. To ensure that the simplified results are compared at the same scale, a verification was performed as follows. First, in every group, each two of three similarity values should be subtracted from each other, and the absolute value should be obtained. Second, for each initial contour line, the absolute values of all groups were averaged (as shown in columns 3 to 5 of Table 2). Third, the maximum value of the three averages was selected and verified to check whether this value is less than 0.002. The max average absolute value (MAAV) of different initial contours was calculated as Equation (11).

$$\begin{aligned}
 MAAV = \text{MAX}(S_{d-ID}(i, DP, CCSM), S_{d-ID}(i, DP, WM), S_{d-ID}(i, WM, CCSM)) \\
 S_{d-ID}(i, A, B) = \text{mean}_{-ID} | \text{Sim}(L_i^A, L_{initial-ID}) - \text{Sim}(L_i^B, L_{initial-ID}) | \quad (11) \\
 (ID = 1, 2, \dots, 20; i = 1, 2, \dots, m)
 \end{aligned}$$

where L_i^A represents the simplified contour generated by the A algorithm, L_i^B represents the simplified contour generated by the B algorithm, $L_{initial-ID}$ represents the initial contour, and i represents the number of groups. By using the above calculation method, the MAAV of the 20 contours was calculated and listed in Table 2. As shown in Table 2, the MAAV was within 0.002 (as shown in the last column of Table 2), thereby guaranteeing that the simplified results produced by the three methods were compared on the same scale.

Table 2. Max average absolute value (MAAV).

Initial Contour ID	Number of Groups	$S_{d-ID}(i, DP, CCSM)$	$S_{d-ID}(i, DP, WM)$	$S_{d-ID}(i, WM, CCSM)$	MAAV
1	4	0.0004	0.0020	0.0020	0.0020
2	2	0.0010	0.0006	0.0005	0.0010
3	2	0.0007	0.0006	0.0013	0.0013
4	3	0.0003	0.0010	0.0013	0.0013
5	4	0.0004	0.0014	0.0012	0.0014
6	4	0.0003	0.0008	0.0011	0.0011
7	2	0.0015	0.0018	0.0013	0.0018
8	2	0.0004	0.0002	0.0005	0.0005
9	2	0.0008	0.0010	0.0018	0.0018
10	4	0.0006	0.0006	0.0009	0.0009
11	4	0.0007	0.0013	0.0024	0.0024
12	3	0.0003	0.0013	0.0015	0.0015
13	2	0.0009	0.0013	0.0020	0.0020
14	2	0.00001	0.0002	0.0002	0.0002
15	4	0.0001	0.0009	0.0010	0.0010
16	4	0.0003	0.0011	0.0009	0.0011
17	2	0.0007	0.0002	0.0007	0.0007
18	4	0.0003	0.0016	0.0019	0.0019
19	2	0.0019	0.0001	0.0020	0.0020
20	2	0.0007	0.0011	0.0007	0.0011

Linear density was used as an indicator for evaluating the ability of each method to retain global structural features [43]. This indicator was defined as Equation (12).

$$\text{Linear Density}(L_i) = \frac{\text{Length}(L_i)}{\text{Area}(\text{Buffer}(L_i))} \quad (12)$$

where $\text{Length}(L_i)$ represents the length of the simplified contour, and $\text{Area}(\text{Buffer}(L_i))$ represents the value of the buffer area of the simplified contour. According to Wu [44], the standard position deviation reflects the global deviation of a simplified line from the

initial line. Therefore, the standard deviation of the position was designated as the width parameter in the buffer operation.

The linear densities were calculated separately for the simplified contours at the same scale in each group of the 20 initial contours. The average values of each initial contour linear density corresponding to the three methods are shown in Figure 11. According to Deng [43], a higher linear density indicates a better overall trend retention rate. As shown in Figure 11, the DP algorithm, WM algorithm, and CCSM show differences for each contour group. The linear density value obtained by CCSM was higher than that obtained by the DP algorithm, indicating that the former has a better global structure retention ability than the latter. Several contour groups of the linear densities of CCSM were significantly better than those of the WM algorithm (such as groups 6#, 8#, 13#, and 15#, etc.). Overall, CCSM is superior to the other two methods in terms of global structural feature retention.

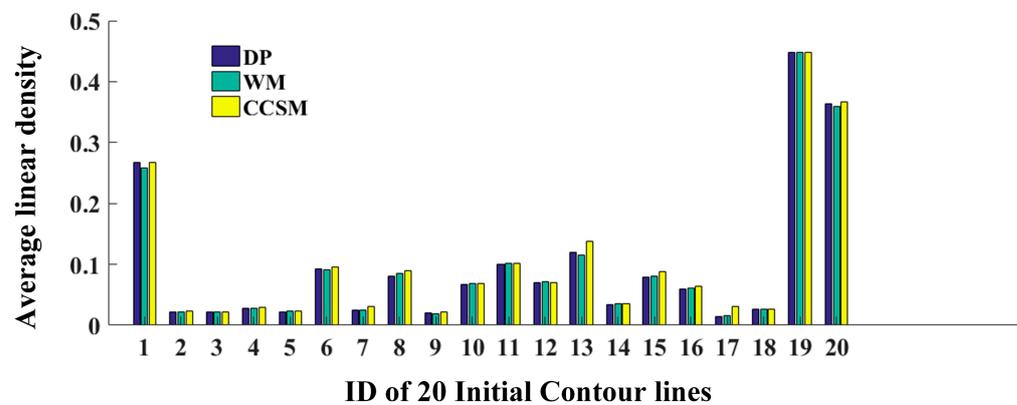


Figure 11. Average linear densities of the 20 selected contours generated by the three methods.

3.2.4. Comparison of Position Change

Generally, simplifying contour lines can trigger changes in shape, and a higher degree of simplification corresponds to a higher amount of distortion. One way of measuring the error induced by simplification is by calculating the location difference between the original and simplified lines. The position change of a curve mainly represents the geometric accuracy of the linear features. The simplified contours generated by the three simplification methods were compared based on the mean, standard, and max differences of position. The change in the position of the simplified contours was calculated by using the computational positional error method proposed by Elmar [45]. For each original point of the initial contour, the positional error was calculated as the perpendicular difference between that point and the corresponding line segment of the simplified contour. The distance d from the point $P_0(x_0, y_0)$ to the corresponding line segment $l: Ax + By + C = 0$ was defined as Equation (13). Applying these simplification algorithms consistently produced low positional errors.

$$d = \frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}} \quad (13)$$

For the validation, we used 20 randomly generated contour lines (described in Section 3.2.3) as experimental data. We compared three position differences, namely, the standard, average, and max position differences, among the simplified results generated by the three simplification methods and the initial contours at the same scale. As shown in Table 3, CCSM obtained the smallest standard positional error (208.199 m) and maximum positional error (778.478 m) among the three simplification methods, thereby suggesting that CCSM can effectively maintain the global shape characteristics of the contours better. Given that the real value of the mean positional error is susceptible to the local extremum, the mean positional error values may not stably reflect the overall displacement [44].

Table 3. Numerical evaluation of the positional errors between the initial and simplified contours obtained by the DP algorithm, WM algorithm, and CCSM at the same scale.

Method	Std (m ^a)	Mean (m)	Max (m)
The DP algorithm	208.297	451.011	778.567
The WM algorithm	208.246	451.158	778.640
CCSM	208.199	451.053	778.478

(^am = meter).

4. Discussion

4.1. Effect of CCS's Mesh Parameters on Similarity

To examine the effect of CCS's mesh parameters on the similarities among simplified contour lines, we used different mesh parameters for CCS's tessellation. First, as shown in Figure 12, the contour with 86 points was taken as the initial contour and was labeled C₁. By using the DP algorithm, C₁ was simplified into a target contour labeled C₂ with only the starting and ending points of C₁ (representing the situation of extreme simplification), whereas C₁ was simplified into a target contour labeled C₃₅ with 35 points of C₁ (representing the situation of middle simplification).

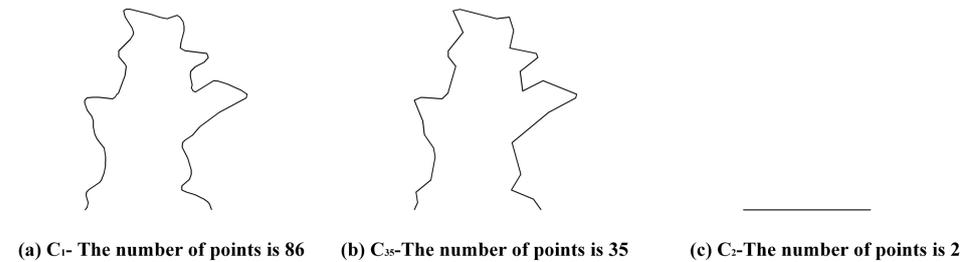


Figure 12. Different contours with different points. (a) The initial contour (C₁) with 86 points. (b) The simplified contour (C₃₅) with 35 points. (c) The simplified contour (C₂) with 2 points.

We then built CCS-2 by using C₁ (initial contour) and C₂ (target contour) and built CCS-35 by using C₁ (initial contour) and C₃₅ (target contour). Both CCS-35 and CCS-2 were established under different mesh parameters (i.e., 0.01, 0.008, 0.006, 0.004, 0.002, and 0.001). Figure 13 shows CCS-35 and CCS-2 under different mesh parameters.

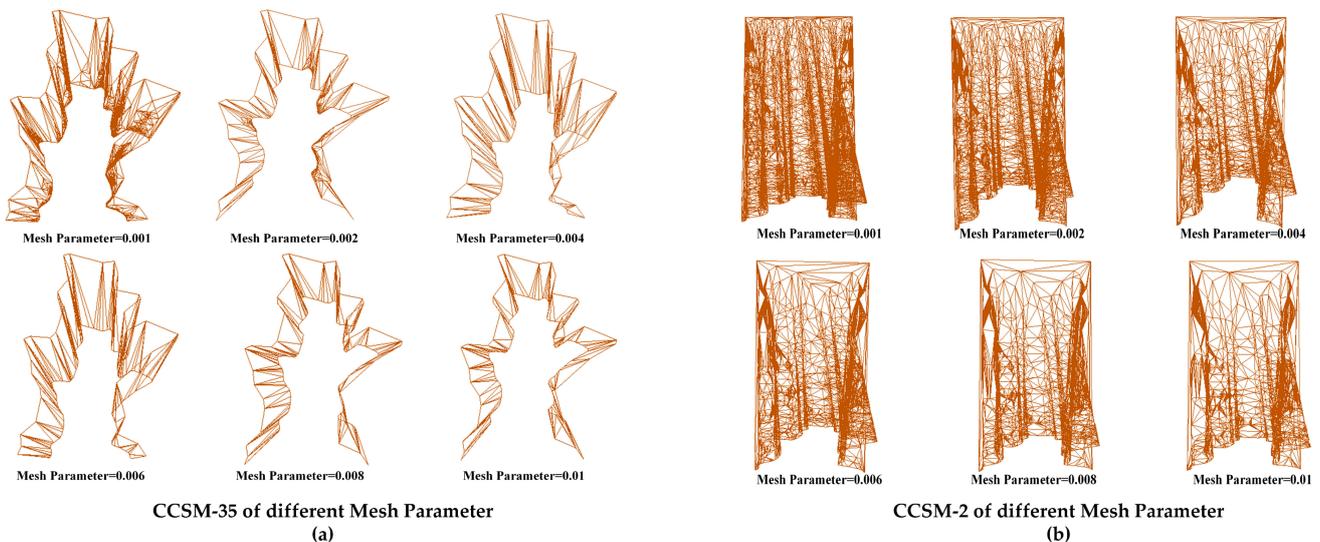


Figure 13. CCS-35 and CCS-2 under different mesh parameters. (a) CCS-35. (b) CCS-2.

We obtained 100 simplified contours by intersecting 100 horizontal planes with CCS-35 and CCS-2 and then calculated the similarities between the simplified contours and C1. Figure 14a shows that when the target contour has 35 points, the trend of the similarity curve corresponding to each mesh parameter is roughly the same. Furthermore, when the mesh parameters were set to 0.001, 0.004, 0.006, and 0.008, these curves were almost similar and overlapping. When the mesh parameters were set to 0.002 and 0.01, some small local differences were observed among the curves, thereby suggesting that despite slight local differences, the curve trends of all mesh parameters were globally consistent. Figure 14b shows that when the target contour contains only the starting and ending points, the mesh parameters show similar and almost overlapping curve trends, which suggests that the similarity between the simplified and initial contours is insensitive to the mesh parameter, that is, mesh parameter does not affect the trend of similarity.

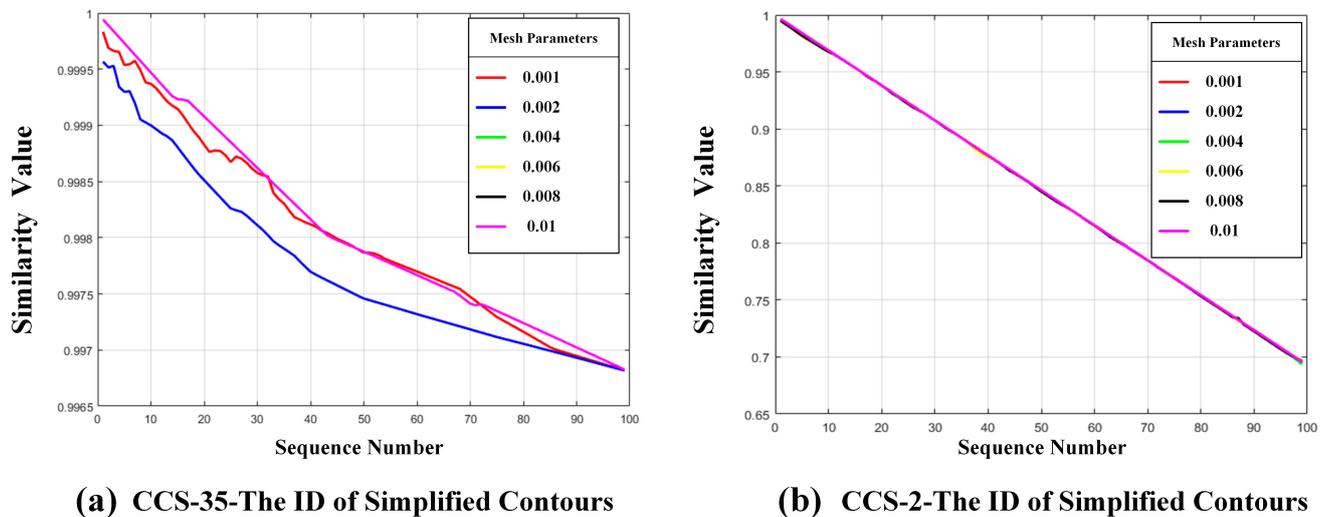


Figure 14. Similarity between the simplified and initial contours with different mesh parameters. (a) CCS-35. (b) CCS-2.

Therefore, the similarity trend is generally consistent under different mesh parameters. The mesh parameters for CCSM have a relatively stable influence on the similarity assessment, especially for a wider scale range of multi-scale line simplifications. Moreover, the CCSM constructed by using a specific target contour has a certain influence on the similarity, but such influence can be alleviated by selecting appropriate mesh parameters. Given that a smaller mesh parameter corresponds to a higher computation cost, selecting a neutral value can balance the mesh size and quality of the solution.

4.2. Effect of Point Number of Target Scale Contour

In CCSM, the shape change of the intermediate simplified contours is related to the initial and target contours. To verify whether the point number of the target scale contour affects the simplified results, two contours with different shapes were randomly selected from the initial contour set. Each target contour had 100, 50, 40, 30, 20, 10, and 2 points as shown in Figure 15a,b. The target contour with only two points was selected given its representativeness of extreme simplification.

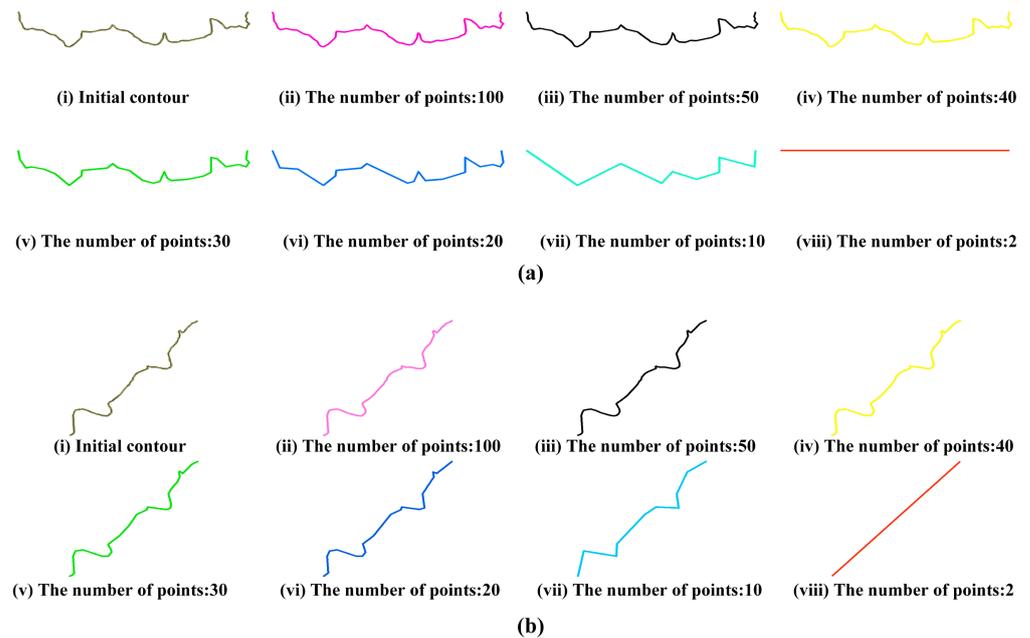


Figure 15. Two selected contours and the corresponding target contours with different point numbers. (i) Initial contour. (ii–viii) Target contours with 100, 50, 40, 30, 20, 10, and 2 points, respectively. (a,b) represent two contours with different shapes randomly selected from the initial contour set.

Figure 16a,b show the relationship between the similarity and scale of these two contours. The similarity here represents the similarity between the simplified and initial contours, whereas scale represents the estimated map scale of the simplified contour. Figure 16 shows that the similarity gradually decreases when the number of points is relatively large and significantly decreases when the number of points reduces to 2. However, regardless of the number of points in the target contour, the trends between scale and similarity were generally very consistent and orderly and demonstrated continuous and monotonous characteristics. Moreover, the simplified contour lines generated by the CCSM were continuous and multi-scale. In other words, our proposed method can be applied to different points of target contours.

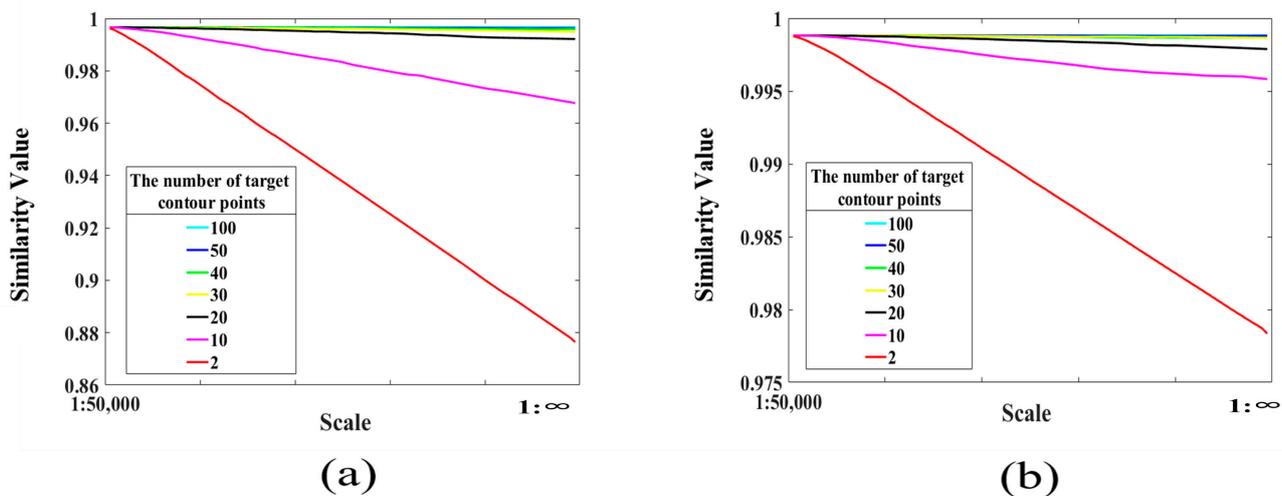


Figure 16. Relationship between the scale and similarity of the target contours at different points. (a) Relationship between the scale and similarity of contours in Figure 15a. (b) Relationship between the scale and similarity of contours in Figure 15b.

4.3. Ability of Arbitrary Scale Simplification

We aim to generate a set of global continuous multi-scale simplified contours based on the initial contour. The advantage of CCSM lies in the fact that the contour lines of the desired scale can be obtained between fixed scales. Therefore, obtaining a simplified contour with the specified scale from the initial contour is critical. Given that CCSM shows a monotonous and continuous trend between scale and similarity as discussed in Section 3.2.2, the similarity between the simplified and initial contours is considered a proxy indicator for obtaining a simplified contour with a specified scale. In other words, similarity can serve as an intermediary to obtain the simplified contour corresponding to an arbitrary desired scale.

According to Shen [35], an exponential regression function can be used to establish an empirical relationship between the similarities and scales of contours. However, such a relationship may be unfit for other types of polylines. The exponential regression function is defined in Equation (14).

$$f(x) = a * \exp(b * x) + c * \exp(d * x) \tag{14}$$

where x represents the scale, $f(x)$ represents the similarity value between two contour lines, and a, b, c, d represent coefficient values in exponential regression functions.

The detailed process is described as follows. First, all contours in the study area were divided into five groups (as shown in Figure 17), and the shape and size of the contour line were used as criteria [16]. Second, one contour in each group was selected as reference data (boldface in Figure 17). Third, for the reference data in each group, the similarity values (Equation (9)) between the initial contour (1:50,000) and the contours of the three other scales (1:100,000, 1:250,000, and 1:500,000 downloaded from Google Earth) were then calculated. Table 4 shows the computation results of reference data in every group.

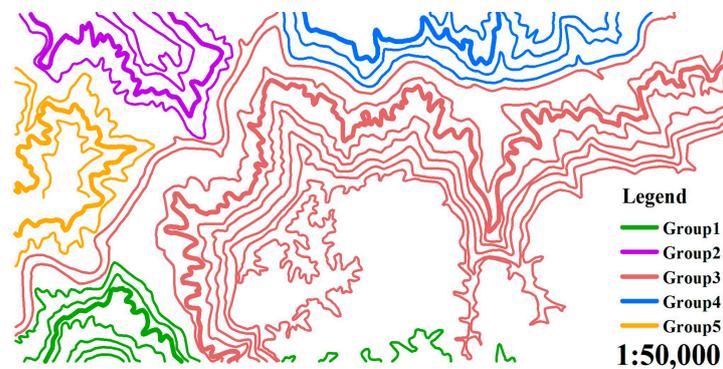


Figure 17. The contour dataset is divided into five groups, with each group assigned a different color. The contours shown in boldface are the reference contours selected from each group.

Table 4. Similarity values between the initial contours and the other scales of contours for reference data in different groups.

Scale	The Similarity of Different Groups				
	Group1	Group2	Group3	Group4	Group5
1:50,000	1	1	1	1	1
1:100,000	0.998	0.988	0.995	0.989	0.997
1:250,000	0.992	0.951	0.986	0.978	0.993
1:500,000	0.978	0.929	0.969	0.947	0.985

Table 4 shows that the exponential regression function coefficients for each group can be fitted by using the values of the known scales and the similarities of the reference

contours in each group. Table 5 shows the coefficient values for each equation in these five groups.

Table 5. Coefficient values of the exponential regression functions in different groups.

Coefficient Name	Coefficient Values of Different Groups				
	Group1	Group2	Group3	Group4	Group5
a	0.998	1.032	0.994	0.988	0.997
b	0.002	−0.01438	0.004	0.009	0.002
c	-9.37×10^{-5}	−0.04805	−0.0002	-2.02×10^{-5}	-3.75×10^{-5}
d	−6.075	−1.02	−5.542	−8.531	−6.361

As for CCSM, the shape similarity–scale trend is continuous and monotonous as shown in Figure 10. Therefore, the scale and similarity value of each simplified result have a one-to-one correspondence. In this case, we can generate simplified contours at any arbitrary scale by using CCSM. A simplified contour of an arbitrary scale can be obtained as follows. First, by using the exponential regression function in every group, the similarity value corresponding to the desired scale can be obtained. Second, by referring to Equation (9), the corresponding Z_i can be deduced from the similarity value. Third, the desired scale simplified contour can be selected based on the value of i . This method provides a potential mechanism for achieving arbitrary scale simplification.

Figure 18 shows the simplified contours with four specified scales (1:100,000, 1:200,000, 1:250,000, and 1:500,000) obtained by CCSM. For most of the simplified contours, the essential shape was unchanged during the progressive change process, thereby suggesting that the overall curved shape was well maintained. Individual contours with a small size disappeared as the scale decreased because their similarity values were filtered out according to the similarity threshold based on the small scale and exponential regression function. This finding is consistent with the rule of discarding small elements in cartographic generalization [21]. Meanwhile, the images in these scales seemed similar because all bends on a single contour line were transformed together (as shown in Figure 9c). This finding also indicates that CCSM performs well in continuous and global deformation in the simplification of contours.

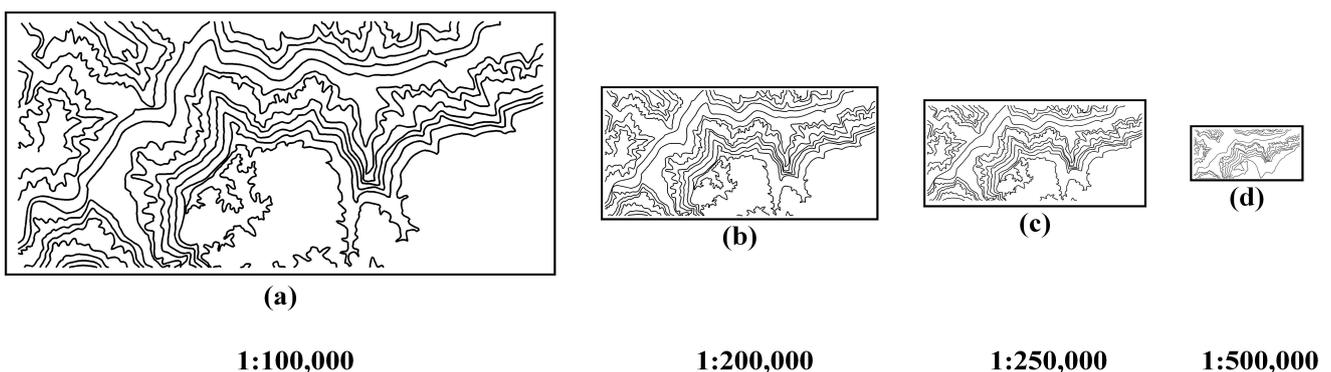


Figure 18. Simplified contours generated by CCSM at the 1:100,000 (a), 1:200,000 (b), 1:250,000 (c), and 1:500,000 (d) scales.

4.4. Topographical Feature Retention

In general, feature retention should match the degree of simplification. From a geographical perspective, the maintenance of terrain features plays an essential role in the simplification process. To verify the capability of CCSM for topographic feature retention, in this subsection, the retention of typical topographic features (e.g., valley lines and ridgelines) in continuous multi-scale simplified contours is evaluated by visual analysis methods. Figure 19 shows the valley lines and ridge lines were extracted from a contour map at a scale of 1:50,000.

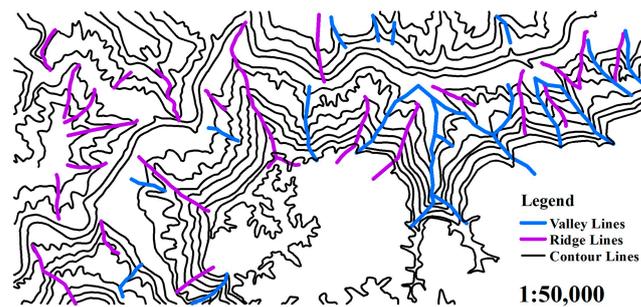


Figure 19. Terrain features in the terrain map of 1:50,000.

As shown in Figure 20, with the progressive simplification of the contour line, some details of the contour lines were continuously reduced, and some ridge and valley lines gradually disappeared. The simplified terrain map only retained some key ridge and valley lines. However, the topological relationship of all simplified contours was also consistent, thereby suggesting that the key corresponding geographical features were maintained. Therefore, CCSM also performs well in maintaining the terrain features of geographical contours with complex linear features. Figure 20 also shows that even though CCSM is used for single line simplification, this method can also well maintain the group multi-scale continuous rendering of simplified contour lines.

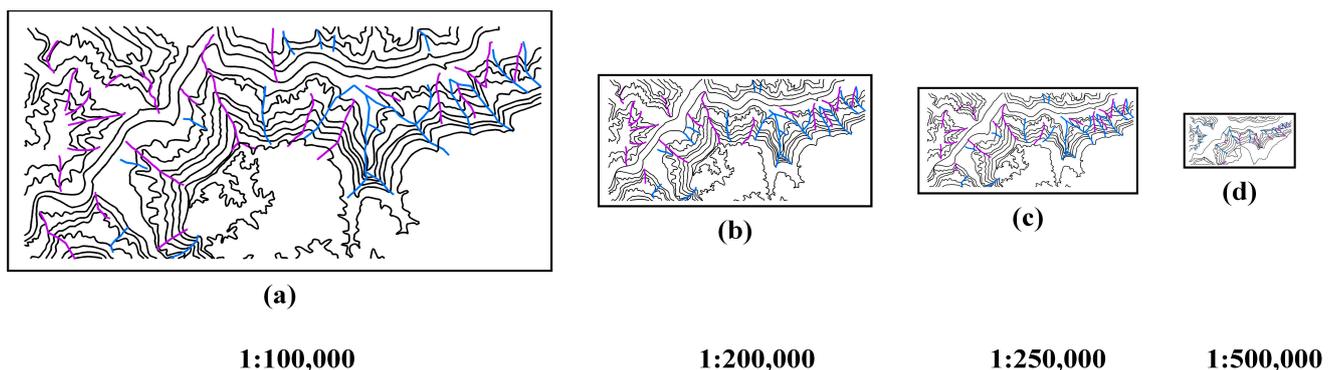


Figure 20. (a–d) Topographical feature retention of the terrain map at the 1:100,000, 1:200,000, 1:250,000, and 1:500,000 scales.

In practice, the target contour L_n is not always easy to obtain or even inexistent at the targeted scale [21]. For the former situation, a general simplification method (i.e., DP algorithm) could be employed to obtain the target contour line L_n [46]. That is, by using the DP algorithm, the significant morphology characteristic points of L_0 could be extracted and linked together to form L_n in an orderly manner. The main topographical feature is retained in L_n . For the later, it is an extreme case due to the principles of cartographic generalization. The starting and ending points of L_0 could be directly linked to form L_n . The main topographical feature is disappeared in L_n . This is consistent with the inexistent status of L_n in the targeted map.

5. Conclusions

This study proposed the CCSM method for contour simplification based on the NURBS surface between the initial and target contours with a feature constraint. A set of horizontal planes with equidistant height values was intersected with CCSM to obtain global continuous multi-scale simplified contours.

The CCSM method proposed in this paper can support continuous multi-scale contour simplification while maintaining the essential global structure. Compared with the classical DP and WM algorithms, CCSM can provide simplified contours with better continuity and consistency. This study also proposed a shape similarity–scale trend line based on

the contour deformation analysis method that can be used to evaluate the performance of multi-scale polyline simplification methods.

Two problems still need to be studied in depth in future work. First, CCSM consumes much time in surface construction, thereby suggesting that this method needs to be optimized further. Second, a more general form of the relationship between the contour scale and the height value of the horizontal plane for an arbitrary type contour should be explored.

Author Contributions: Conceptualization, Z.Z., L.Y., R.W., and H.Q.; methodology, Z.Z. and H.Q.; software, H.Q.; validation, H.Q. and Z.Z.; formal analysis, Z.Z., L.Y., and H.Q.; investigation, H.Q.; resources, Z.Z.; data curation, H.Q. and W.Z.; writing—original draft preparation, H.Q.; writing—review and editing, Z.Z., L.Y., R.W., and H.Q.; visualization, H.Q. and Z.Z.; supervision, S.Z.; project administration, S.Z.; funding acquisition, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Land and Resources (KF-2018-03-038).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and code that support the findings of this study are available in “figshare.com” with the private link <https://figshare.com/s/c54a8fd8c787ef3f913a> (accessed on 22 July 2019). We provided the method of the CCSM and the contour dataset. The method can be run using the software “Visual Studio 2017”.

Acknowledgments: The authors would like to thank the anonymous reviewers for their valuable comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Liu, P.; Xiao, T.; Xiao, J. A multi-scale representation model of polyline based on head/tail breaks. *Int. J. Geogr. Inf. Sci.* **2020**, *34*, 2275–2295. [[CrossRef](#)]
- Van Oosterom, P. Variable-scale Topological Data Structures Suitable for Progressive Data Transfer: The GAP-face Tree and GAP-edge Forest. *Cartogr. Geogr. Inf. Sci.* **2005**, *32*, 331–346. [[CrossRef](#)]
- Wu, F.; Zhang, Q.; Gong, X. Matching and Classification Model for Multi-Scale Transformation and Representation of Spatial Data. *J. Geomat. Sci. Technol.* **2014**, *4*, 331–335.
- Li, Z.; Zhou, Q. Integration of linear and areal hierarchies for continuous multi-scale representation of road networks. *Int. J. Geogr. Inf. Sci.* **2012**, *26*, 855–880. [[CrossRef](#)]
- Weibel, R. Models and Experiments for Adaptive Computer-Assisted Terrain Generalization. *Cartogr. Geogr. Inf. Sci.* **1992**, *19*, 133–153.
- Hao, Z.; Li, C.; Yin, Y.; Wu, P.; Wu, W. A contour interpolation algorithm based on Fréchet distance. *Bull. Surv. Mapp.* **2019**, *1*, 65–68.
- Itzhak, E.; Yoeli, P.; Doysther, Y. Analytic Generalization of Topographic Data and Terrain Models. In Proceedings of the 22nd ICA International Cartographic Conference, Coronary, Spain, 11–16 July 2005.
- Ai, T.; Ke, S.; Yang, M. Envelope generation and simplification of polylines using Delaunay triangulation. *Int. J. Geogr. Inf. Sci.* **2017**, *31*, 297–319. [[CrossRef](#)]
- Li, J.; Ai, T.; Liu, P. Continuous Scale Transformations of Linear Features Using Simulated Annealing-Based Morphing. *Isprs. Int. J. Geo-Inf.* **2017**, *6*, 242. [[CrossRef](#)]
- Douglas, D.H.; Peucker, T.K. Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature. *Can. Cartogr.* **1973**, *10*, 112–122. [[CrossRef](#)]
- Visvalingam, M.; Whyatt, J. Line generalisation by repeated elimination of points. *Cartogr. J.* **1993**, *30*, 46–51. [[CrossRef](#)]
- Saalfeld, A. Topologically Consistent Line Simplification with the Douglas-Peucker Algorithm. *Cartogr. Geogr. Inf. Sci.* **1999**, *26*, 7–18. [[CrossRef](#)]
- Wu, S.; Marquez, M.R. A non-self-intersection Douglas-Peucker algorithm. In Proceedings of the 16th Brazilian symposium on computer graphics and image processing, Sao Carlos, Brazil, 12–15 October 2003.
- Wu, S.; Silva, A.C.; Marquez, M.R. The Douglas-peucker algorithm: Sufficiency conditions for non-self-intersections. *J. Braz. Comput. Soc.* **2004**, *9*, 67–84. [[CrossRef](#)]
- Pallero, J.L.G. Robust line simplification on the plane. *Comput. Geosci.* **2013**, *61*, 152–159. [[CrossRef](#)]

16. Qian, H.; Zhang, M.; Wu, F. A New Simplification Approach Based on the Oblique-Dividing-Curve Method for Contour Lines. *ISPRS Int. J. Geo. Inf.* **2016**, *5*, 153. [[CrossRef](#)]
17. Jones, C.B. Map generalization with a triangulated data structure. *Cartogr. Geogr. Inf. Sci.* **1995**, *22*, 317–331.
18. Wang, Z.; Muller, J. Line Generalization Based on Analysis of Shape Characteristics. *Cartogr. Geogr. Inf. Sci.* **1998**, *25*, 3–15. [[CrossRef](#)]
19. Mustafa, N.; Krishnan, S.; Varadhan, G. Dynamic simplification and visualization of large maps. *Int. J. Geogr. Inf. Sci.* **2006**, *20*, 273–302. [[CrossRef](#)]
20. Zhu, Q.; Wu, F. A Novel of Contour Line Simplification Algorithm Base on Visibility. *J. Imag. Graph.* **2009**, *14*, 359–364.
21. Li, Z.; Openshaw, S. A Natural Principle for the Objective Generalization of Digital Maps. *Cartogr. Geogr. Inf. Sci.* **1993**, *20*, 19–29. [[CrossRef](#)]
22. Bertolotto, M.; Egenhofer, M. Progressive Transmission of Vector Map Data over the World Wide Web. *Geoinformatica* **2001**, *5*, 345–373. [[CrossRef](#)]
23. Deng, M.; Peng, D. Morphing Linear Features Based on Their Entire Structures. *Trans. GIS* **2015**, *19*, 653–677. [[CrossRef](#)]
24. Du, J. A Progressive Method of Simplifying Polylines with Multi-bends Simplification Templates. In Proceedings of the CCF Chinese Conference on Computer Vision, Tianjin, China, 11–14 October 2017; pp. 515–528.
25. Nöllenburg, M.; Merrick, D.; Wolff, A. Morphing polylines: A step towards continuous generalization. *Comput. Environ. Urban Syst.* **2008**, *32*, 248–260. [[CrossRef](#)]
26. Osher, S.; Sethian, J. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.* **1988**, *79*, 12–49. [[CrossRef](#)]
27. Timp, S.; Karssemeijer, N. A new 2D segmentation method based on dynamic programming applied to computer aided detection in mammography. *Med. Phys.* **2004**, *31*, 958–971. [[CrossRef](#)] [[PubMed](#)]
28. Piegl, L. On NURBS: A survey. *IEEE Comput. Graph. Appl.* **1991**, *11*, 55–71. [[CrossRef](#)]
29. Hu, S.M. Modifying the shape of NURBS surfaces with geometric constraints. *Comput. Aided. Des.* **2001**, *33*, 903–912. [[CrossRef](#)]
30. Li, Z.; Wang, J.; Tan, S.; Xu, Z. Scale in Geo-information Science: An Overview of Thirty-year Development. *Geomat. Inf. Sci. Wuhan. Univ.* **2018**, *43*, 480–489.
31. Li, W.; Hsu, C.Y. Automated terrain feature identification from remote sensing imagery: A deep learning approach. *Int. J. Geogr. Inf. Sci.* **2018**, *34*, 637–669. [[CrossRef](#)]
32. Ahn, H.K.; Knauer, C.; Scherfenberg, M. Computing the Discrete Fréchet Distance with Imprecise Input. *Int. J. Comput. Geom. Appl.* **2012**, *22*, 422–433. [[CrossRef](#)]
33. Eiter, T.; Mannila, H. *Computing Discrete Fréchet Distance*; Technical Report CD-TR 94/64; Christian Doppler Laboratory for Expert Systems: Vienna, Austria, 1994; Volume 64, pp. 636–637.
34. Aronov, B.; Har-Peled, S.; Knauer, C. Fréchet Distance for Curves, Revisited. In Proceedings of the European Symposium on Algorithms, Zurich, Switzerland, 11–13 September 2006.
35. Wolin, A.; Eoff, B.; Hammond, T. ShortStraw: A simple and effective corner finder for polylines. In Proceedings of the Fifth Eurographics Conference on Sketch-Based Interfaces and Modeling 2008, Annecy, France, 11–13 June 2008; p. 3340.
36. Shen, L.; Ai, T.; Li, J.; Huang, L.; Li, W. A progressive method for the collapse of river representation considering geographical characteristics. *Int. J. Digit. Earth* **2020**, *13*, 1366–1390. [[CrossRef](#)]
37. Kriegeskorte, N. Representational similarity analysis-connecting the branches of systems neuroscience. *Front. Syst. Neurosci.* **2008**, *2*, 4. [[CrossRef](#)]
38. Owen, J. *STEP: An Introduction*, 2nd ed.; Information Geometers: Winchester, UK, 1997; pp. 210–215.
39. Shi, L.; Zhao, B.; Li, J.; Zhang, L. An analysis of Google Earth elevation data accuracy and its application to seismic exploration. *Geophys. Geochem. Explor.* **2016**, *40*, 578–586.
40. Loncaric, S. A survey of shape analysis techniques. *Pattern Recogn.* **1998**, *31*, 983–1001. [[CrossRef](#)]
41. Zhang, P. A shape similarity based change detection approach of multi-resolution remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2012**, *1*, 263–266. [[CrossRef](#)]
42. Liu, Z.; Ai, T. Responsively Simplifying Line by Adapting to Display Environment. *J. Geomat.* **2017**, *42*, 78–82.
43. Deng, M.; Fan, Z.; Liu, H. Performance Evaluation of Line Simplification Algorithms Based on Hierarchical Information Content. *Acta. Geod. Geophys.* **2013**, *42*, 767–773.
44. Wu, F.; Zhu, K. Geometric accuracy assessment of linear feature’s simplification algorithms. *Geo Spat. Inf. Sci.* **2008**, *06*, 600–603.
45. Polyline Simplification. Available online: <https://www.codeproject.com/Articles/114797/PolylineSimplification#headingNP> (accessed on 4 June 2020).
46. Xie, Z.; Wang, H.; Wu, L. The Improved Douglas-Peucker Algorithm Based on the Contour Character. In Proceedings of the 19th International Conference on Geoinformatics, Shanghai, China, 24–26 June 2011.