姚砺, 权泰宇, 万燕. 基于重要值排序和自适应阈值的 Douglas-Peucker 算法[J]. 智能计算机与应用,2024,14(5):101-106.
DOI:10.20169/j.issn.2095-2163.240513

基于重要值排序和自适应阈值的 Douglas-Peucker 算法

姚 砺, 权泰宇, 万 燕
(东华大学 计算机科学与技术学院, 上海 201620)

摘 要: 矢量数据压缩长久以来一直是地理信息系统(GIS)领域的关注焦点,旨在缩减数据规模以满足应用性能方面的需求,同时减少数据传输、系统处理时长以及储存成本,从而提高系统性能和运行开销。虽然在该问题上已取得了一些研究成果,但由于技术进步和新需求的涌现,对矢量数据压缩的压缩率和精度都提出了更高的要求,同时如何确定最优的阈值也成为了一个亟待解决的问题。因此,本文从矢量数据中不同节点对整个矢量图形产生变化的重要性以及如何确定最佳阈值出发,设计了结合重要值排序和自适应阈值的 Douglas-Peucker 算法。通过对上海市民政部门内部矢量数据集的实验表明,改进算法在压缩率相同情况下,数据压缩效果整体优于 Douglas-Peucker 算法及其改进算法。

关键词: 矢量数据压缩; 节点重要值排序; 自适应阈值
中图分类号: P208 **文献标志码**: A **文章编号**: 2095-2163(2024)05-0101-06

Douglas-Peucker algorithm based on important value sorting and adaptive threshold

YAO Li, QUAN Taiyu, WAN Yan
(School of Computer Science and Technology, Donghua University, Shanghai 201620, China)

Abstract: Vector data compression has long been a focal point in the Geographic Information System (GIS) field, aiming to reduce data size to meet application performance requirements, while also minimizing data transmission, system processing time and storage costs, ultimately enhancing system performance and operational efficiency. Although some research results have been achieved in this problem, higher requirements for compression rate and accuracy have been put forward due to technological progress and new demands. At the same time, how to determine the optimal threshold has become a problem to be solved. Therefore, this paper proposes the Douglas-Peucker algorithm based on importance ranking and adaptive threshold from the importance of different nodes in vector data that cause changes in the entire vector graphics and how to determine the optimal threshold. The experiment on the internal vector dataset of the Shanghai Civil Affairs Bureau shows that, under the same compression rate, the improved algorithm has an overall better data compression effect than both the Douglas-Peucker algorithm and its improved algorithms.

Key words: vector data compression; node important value sorting; adaptive threshold

0 引 言

GIS 中的矢量数据概念,无论是何种类型的元素,都是以点坐标的形式包含了所有元素的基本组成信息[1-3]。大部分矢量数据缩减方法都依赖特定算法提取关键节点,旨在保留向量图像整体形态的基础上减少数据点数量,进而缩短数据传输时长并节省存储空间。多年来,国内外学者提出了很多矢量数据的压缩算法,包括角度法[4-6]、Douglas-Peucker 算法[7]、Visvalingam-Whyatt 算法[8]、Li-Openshaw 算法[9]、基于小波分析的压缩算法[10]等。同时智能算法也在近些年大放异彩,在寻找最佳阈值参数方面具有较大优势,例如遗传算法[11]、粒子群算法[12]、狮群算法[13]等。

在上述这些压缩算法中,应用最广泛的就是 Douglas-Peucker 算法(以下简称 DP 算法),DP 算法作为矢量数据压缩领域的经典算法,虽然提出的时间较早,但该算法原理清晰易懂,得到的压缩结果

作者简介: 姚 砺(1967-),男,博士,副教授,硕士生导师,主要研究方向:软件测试技术; 万 燕(1970-),女,博士,教授,硕士生导师,主要研究方向:图像处理,纤维的自动识别。
通讯作者: 权泰宇(1997-),男,硕士研究生,主要研究方向:软件系统开发。Email:Kwontaewoo@163.com

收稿日期: 2023-04-26

具有平移旋转不变性,每个距离阈值与压缩结果一一对应,具有结果唯一性,使得该算法在近些年依然被广泛应用于各种领域,并被不断改进以获得更优秀的压缩效果[14-15]。费若男等学者[16]提出了用偏离方差作为阈值控制手段来改进 DP 算法,将其应用在消光系数边界值的求解上。陈信强等学者[17]提出了将快速捆绑包聚类算法融入 DP 算法,实现对水上交通模式的识别。何宽等学者[18]提出了基于逐点前进法的改进 DP 算法。陈万利等学者[19]将闵式距离融入 DP 算法中,实现对电池健康状况的高精度预测。王荣等学者[20]根据不同地理特征区域采用不同阈值从而实现 DP 算法的全自动化简。周腾等学者[21]对 DP 算法的距离计算方式进行了改进与优化。然而 DP 算法的阈值需要基于经验进行人为设置,无法快速且准确地找到最优阈值。

上述各种算法对矢量数据的压缩依然停留在对压缩率和压缩精度上进行取舍。如果想要更高的压缩率,必然意味着降低压缩后矢量图形形状的精度;如果想要更好地维持矢量图形形状,那就要降低压缩率。如何在保持压缩率和压缩精度的平衡下,对矢量数据的压缩进行改进,成为了一个亟需解决的问题。同时由于 DP 算法、VW 算法等主流压缩算法都需要设置阈值,对于如何寻找最佳阈值也成为了一个需要深入探讨研究的问题。

基于前文论述,本文设计了一种基于重要性排序的自适应阈值 DP 算法。该算法首先利用节点的局部垂比弦值来量化每个节点的重要性值,分析对图形形状保持贡献较大的关键点。然后,从全局对重要性值进行排序,选出一定比例的较大重要值顶点,作为重要节点。最后,利用自适应阈值对相邻 2 个关键点之间的线段进行压缩,根据每段子区间内的局部特征自适应设置不同的阈值,从整体到局部对 DP 算法进行优化。实验结果表明,改进算法在压缩率相同情况下,数据的压缩效果整体优于 Douglas-Peucker 算法及其改进的其他算法。

1 自适应阈值的重要值排序 DP 算法

随着互联网近年来的发展,数据也呈现出爆发式的增长,体现在 WebGIS 领域中就是矢量数据过于庞大,前端请求矢量数据所耗费的网络传输时间大幅增加,因此需要将矢量数据进行压缩。然而压缩结果的精度与压缩率元无法兼顾,因而如何确定最佳的阈值也成为了目前广泛讨论的热点问题。节点的重要值是用来衡量该节点在保持矢量图形形状的重要性,重要值大表示该节点不可或缺,重要值小则表示该节点一定程度上可以被替代[22]。然而重要值是一个局部概念,不能作为全局的最终压缩结果,因此本文创新性地将重要值进行全局排序,然后取最大的这些节点作为划分区间的依据,并对这些划分的区间进行独立的 DP 算法处理。同时每个区间有自己的地理特征,距离阈值不能统一设置,且距离阈值的设定一般情况下是根据历史经验来人为设定,因此本文创新性将自适应阈值加入到重要值 DP 算法中,对不同区间、不同地形自动计算其最佳阈值,使得压缩数据的压缩率和偏移距离均取得了更好的效果。

1.1 节点重要值

图形上的每个节点对于该图形的形状和结构的影响是不同的,也就是每个节点在维持图形形状上起到的的作用往往并不一样。有些节点具有更大的重要性,如果把这些节点删除,那么该图形会产生较大的形变。衡量图形上一个节点的重要性程度时,可以使用其相邻的 2 个节点来进行局部判断。具体来说,如果从矢量数据所组成的折线中移除一个节点,并将该节点两侧的节点相连形成新的折线,被移除的点距离两侧节点所在直线的距离将会成为衡量该节点对折线形状变化产生影响的重要指标。节点重要值计算方法如图 1 所示,移除节点 P_i 后,折线 $P_{i-1}\,P_i\,P_{i+1}$ 被压缩为折线 $P_{i-1}\,P_{i+1}$,偏移距离为 P_i 到 $P_{i-1}P_{i+1}$ 的垂直距离。此外,更大的 $P_{i-1}P_{i+1}$ 又会削弱垂直距离对线段形状的控制效果。

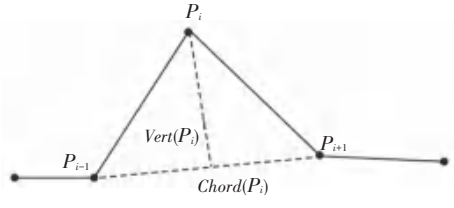

图 1 节点重要值计算方法

Fig. 1 Calculation method of nodes importance value

为此,重要值的公式需要包含 P_i 到 $P_{i-1}\,P_{i+1}$ 的垂直距离 $Vert(P_i)$ 与 $P_{i-1}\,P_{i+1}$ 的距离 $Chord(P_i)$,按上述分析可得,$Vert(P_i)$ 越大、重要值越大,$Chord(P_i)$ 越大、重要值却越小,故对于矢量图形中的任意节点 P_i,其重要值由式(1)来求得:

$$importance(P_i) = \frac{Vert(P_i)}{Chord(P_i)} \tag{1}$$

其中,$P_{i-1}\,P_{i+1}$ 的距离为 $Chord(P_i)$,P_i 到其相邻节点连线的垂直距离为 $Vert(P_i)$。

1.2　重要值排序筛选候选关键点

节点的重要值衡量了该节点对整个矢量图形维持原形状的重要性，重要值越大，该节点对原图形形状越重要。为了使压缩后的矢量图形尽可能地维持原图形形状，就需要保留这些重要值很大的节点。使用节点重要值序列的极大值点作为候选关键点的问题在于只能在局部对图形形状进行约束，最后压缩的结果在整体上并不是最优解。本文通过对节点重要值从大到小排序，选择一定数量重要值足够大的节点作为候选关键点，在全局范围内保持了矢量图形原有的形状。

基于重要值排序的 DP 算法具体步骤如下：

(1)节点重要值的计算。通过式(1)计算矢量数据集中每个节点的重要值；

(2)筛选候选关键点。将所有节点的重要值进行排序，选出最大的前 n 个节点作为候选关键点，给整个压缩后的图形定形状基调；

(2)设置距离阈值 T；

(4)合并候选关键点并生成关键点集。根据 T，对候选关键点集进行 DP 算法处理，生成关键点集合；

(5)对关键点集分段进行 DP 压缩并生成结果集。对关键点集中的所有相邻关键点之间的所有原始节点进行 DP 算法处理，得到压缩后的矢量数据集。

1.3　自适应阈值 DP 算法

在传统 DP 算法中，需要人为设置距离阈值 T，导致需要测试多次才能找到这个矢量数据集的合适阈值，过程繁琐且最终不一定能找到最佳阈值。自适应阈值 DP(ADP)算法通过阈值变化率来确定矢量数据集的距离阈值和关键节点，不再依赖人工设置距离阈值，使得最佳阈值的确定更加容易，这是传统 DP 算法无法做到的[23]。本文创新性将自适应阈值应用到重要值 DP 算法中，相比于重要值排序 DP 算法和原始 DP 算法均在压缩效果上有更好的表现。

在对矢量数据进行压缩时，矢量图形形状变化率高的区域会被显著压缩[24]。对于变化率下降到某一值以下的范围，轨迹保持相对稳定。由此引入了阈值变化率的概念。

(1)阈值变化率 k：在矢量数据被压缩时，如果一个距离阈值能够使这段轨迹中距离基线最远的点在压缩后成为关键点，则该阈值称为该点以及这个矢量轨迹段的临界阈值，记为 ξ。

每段矢量区间可以分为 2 个子区间，设轨迹上所有节点的临界阈值集合为 $M=\{\xi_1,\xi_2,\cdots,\xi_p,\cdots,\xi_n\}$。其中，$\xi_p$ 为第 p 次压缩的临界阈值，n 为临界阈值的个数，相邻 2 个 ξ 之间的步长为 1，则第 p 个临界阈值的临界阈值变化率为：

$$k_p=\begin{cases}|\xi_p-\xi_{p+1}|, & p\leqslant n-1\\ \infty, & p=1\\ 0, & p=n\end{cases} \tag{2}$$

(2)最佳阈值变化率 k_0：当 $k_p\leqslant k_0<k_{p-i}(i=1,2,\cdots,p-1)$ 时，由临界阈值 $\{\xi_1,\xi_2,\cdots,\xi_{p-1}\}$ 对应的点(称为关键点)组成的轨迹在保持高压缩率的情况下，能与原轨迹保持较高的相似性。

对于矢量数据压缩，应该同时兼顾压缩率和结果偏移距离。一般情况下，压缩率越高，偏移距离就越大，压缩结果越失真。因此，需要找到一个阈值，使得压缩率和偏移距离能够平衡。通过观察大量矢量数据可以发现，图形中关键点的 ξ 一般要大于大多数非关键点的 ξ 值。本文使用的数据为作者开发的民政局区划系统项目中的矢量数据集，图 2 为该数据集的各关键点对应临界阈值 ξ 的变化曲线。

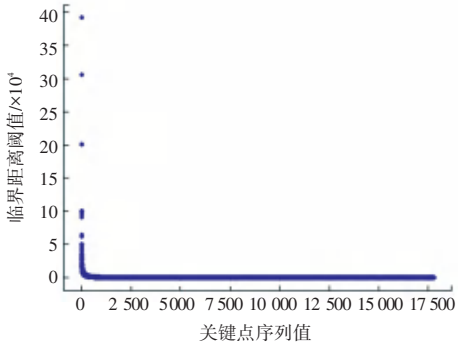

图 2　关键点的临界阈值

Fig. 2　Critical threshold for key points

由图 2 可以看出，阈值的整体变化趋势与反比例函数相似，故设拟合曲线函数为：

$$y=\frac{a}{(bx+c)^d} \tag{3}$$

对图 2 阈值数据进行拟合，得到 4 个参数分别为：$a=84.007\,2$，$b=0.000\,2$，$c=0.000\,3$，$d=1.043\,9$。

对式(3)求导，可得：

$$y'=-\frac{abd}{(bx+c)^{d+1}} \tag{4}$$

角度变化率 r 指的是拟合曲线上相邻两点切线方向上角度的变化率，可由式(5)来表示：

$$r_p=|\arctan y'_p-\arctan y'_{p+1}| \tag{5}$$

可以看到靠近 y 轴的区域临界阈值下降得非常快,斜率几乎不变,靠近 x 轴附近的区域临界阈值变化不明显,斜率也变化不明显。在这 2 个区域中间的位置,也就是图 3 所展示的区域,斜率发生了快速的变化。因此,可以根据拟合曲线中 2 个相邻点之间的角度变化率来计算最优阈值变化率。也就是具有最大斜率变化率的点相对应的阈值变化率就是最优阈值变化率k_0。

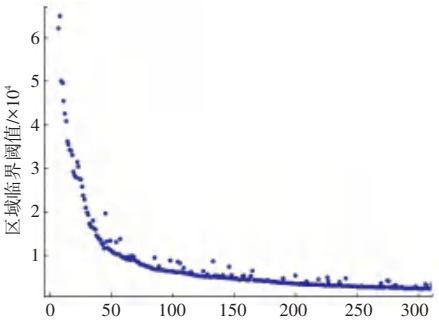

图 3 放大局部区域

Fig. 3 Enlargement on a local area

根据曲线角度变化率的变化(如图 4 所示),当序列号为 579 时,存在一个理论上最优的阈值变化率、即0.001 537。

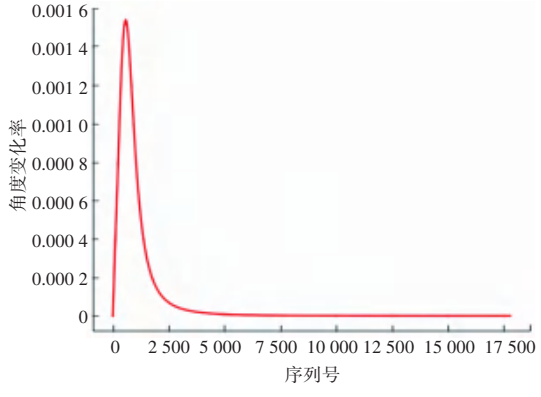

图 4 角度变化率

Fig. 4 Change rate of angle

ADP 的压缩过程如图 5 所示。图 5 中, Set_{origin} 表示初始输入压缩轨迹,Set_{keypoint} 表示经过压缩后得到最终关键点集合,关键点对应的子矢量区间为 $ST_{p,q}(q \leqslant 2)$(表示第 p 次压缩得到的第 q 个子轨迹段),并计算出$ST_{p,q}$ 中关键点$keyPoint_{p,q}$ 对应的临界阈值$\xi_{p,q}$。Set_{ST}、Set_{keypoint} 和Set_{ξ} 分别表示压缩后的子轨迹段$ST_{p,q}$、关键点 $keyPoint$ 和 $keyPoint$ 对应的临界阈值ξ_p 的集合。若$k_p \geqslant k_0$ 且$\xi_p \geqslant T$,则取Set_{ST} 中最大ξ_p 对应的$ST_{\max(\xi)}$ 作为下一个输入的压缩轨迹段,删除Set_{ST} 中的$ST_{\max(\xi)}$;否则,循环结束。

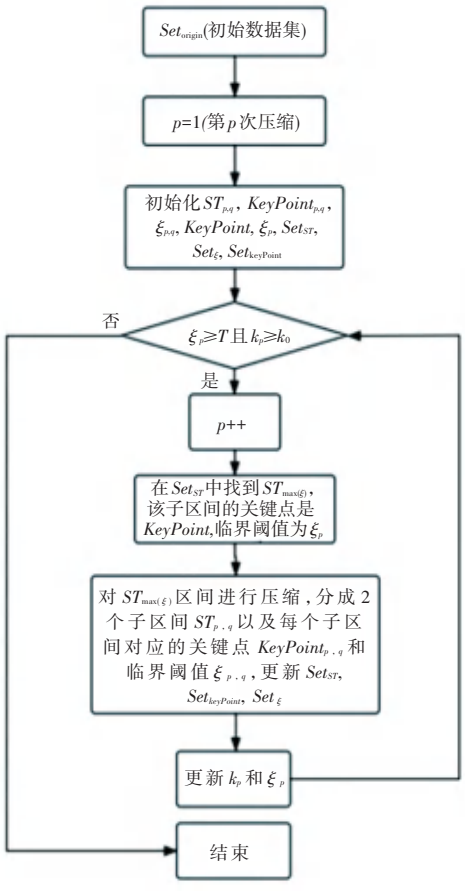

图 5 自适应阈值 DP 算法流程

Fig. 5 Adaptive threshold DP algorithm process

2 基于重要值排序和自适应阈值的 DP 算法流程

(1)节点重要值的计算:通过式(1)计算矢量数据集中每个节点的重要值;

(2)筛选候选关键点:将所有节点的重要值进行排序,选出最大的前 n 个节点作为候选关键点,给整个压缩后的图形定整体形状基调;

(3)设置距离阈值 T;

(4)合并候选关键点并生成关键点集:根据距离阈值 T, 对候选关键点集进行 DP 算法处理,生成关键点集合;

(5)对关键点集分段 ADP:每 2 个重要点之间用 ADP 算法处理原始数据;

(6)计算子集内的所有临界阈值 ξ, 并根据其来求得曲线拟合函数;

(7)计算该分段内最优阈值变化率 k_0, 将该分段放入分段集合 Set_{ST} 中;

(8)初始化 k 和 ξ 为正无穷;

(9)如果 k 大于k_0,且 ξ 大于 T,则继续如下循

环，否则结束循环；

（10）取出 Set_{ST} 中最大的 ξ 对应的 $ST_{\max(\xi)}$ 对象；

（11）将 $ST_{\max(\xi)}$ 对应的关键点加入到结果数据集 Set_{keypoint} 中；

（12）将该分段分成 2 个子区间，分别加入到 Set_{ST} 中；

（13）计算出 ξ_p 和 k_p；

（14）重复第（9）步循环。

3　实验和结果

3.1　实验环境和数据集

本文的实验环境基于 Windows11 操作系统，系统内存为 16 G，CPU 为 i5-12500H。基础实验环境为 Python3.6，Scipy1.9.3，Sympy1.11.1，Matplotlib = 3.6.2。本文实验中使用的数据集为自行开发的上海市民政局行政区划项目中的矢量数据集，由 17 782 个原始数据点组成。

3.2　实验结果比较和分析

本文就 DP 算法和改进算法的具体性能进行了对比分析。为了衡量算法的性能，在相同压缩率下，以压缩数据的偏移距离作为评价标准。压缩率是指压缩后的点与原始数据集的比值，偏移距离定义为原始数据集上每个节点与其所在压缩线段上对应点的距离之和。

表 1 给出了 DP 算法和改进算法在不同压缩比和阈值下矢量数据的实验压缩结果。通过对比结果表明，改进算法取得了比 DP 算法和极大值 DP 算法更好的压缩效果和偏移距离。

表 1　矢量数据压缩实验结果表

Table 1　Vector data compression experiment results table

Compression Rate/%	Displacement Distance		
	DP	Importance DP	Maximal Importance ADP(Ours)
16.477 3	13.180 4	13.152 4	13.145 4
24.806 1	26.338 6	26.322 2	26.244 5
48.014 8	73.188 4	73.127 5	73.090 8
55.663 1	97.531 8	97.670 4	97.247 2
64.649 6	131.261 3	132.124 1	131.131 5
70.329 5	157.968 2	159.698 6	157.700 2
84.152 5	279.663 6	279.808 5	279.566 1
90.940 3	355.901 2	356.614 4	355.757 7

实验结果如图 6 所示。在图 6 中，图 6（a）和图 6（c）是采用传统改进的 Importance DP 算法，图 6（b）和图 6（d）是本文的改进算法。图 6（a）和图 6（b）为宏观整体角度，图 6（c）和图 6（d）为具体到某一区间的矢量数据点。可以看出，在整体上压缩效果大体相似，但是放大到细节时，可以发现本文的算法会删除平缓区域的点（黄色框所示区域），并在崎岖弯折的区域提供更多的数据点用以维持图形形状不变（绿色框所示区域）。因此可以在保持整体压缩率不变的情况下降低压缩的偏移距离。

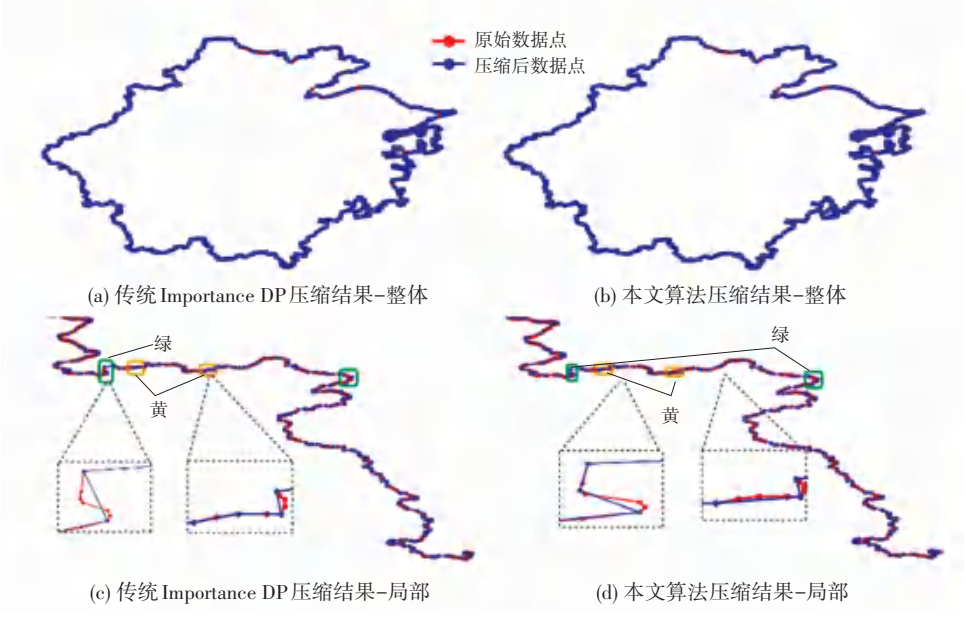

图 6　实验结果

Fig. 6　Experimental results

4 结束语

针对矢量数据压缩存在压缩率和压缩精度不能满足日益增长的 GIS 使用需求、无法针对矢量图形中不同区域的特性进行自适应设置阈值等情况，本文提出采用排序算法从全局角度筛选重要值大的节点作为关键点，对矢量图形划分出不同区域，并对每个区域采用自适应阈值算法，自动计算出与该区域图形形状相符合的最佳阈值。本文提出的方法能有效提升压缩后的精度，高于 DP 算法和其他相关改进算法。

参考文献

[1] 田鹏，郑扣根，潘云鹤. 基于 Strip-Tree 的无级比例尺 GIS 多边形化简技术[J]. 软件学报，2001，12(10)：1495-1502.

[2] SONG J，MIAO R. A Novel Evaluation approach for line simplification algorithms towards vector map visualization [J]. ISPRS International Journal of Geo-Information，2016，5(12)：223.

[3] 杜佳威，武芳，李靖涵，等. 采用多元弯曲组划分的线要素化简方法[J]. 计算机辅助设计与图形学学报，2017，29(12)：2189-2196.

[4] MCMASTER R B. A statistical analysis of mathematical measures for linear simplification[J]. The American Cartographer，1986，13(2)：103-116.

[5] MCMASTER R B. Automated line generalization[J]. Cartographica：The International Journal for Geographic Information and Geovisualization，1987，24(2)：74-111.

[6] LI Zhilin. An examination of algorithms for the detection of critical points on digital cartographic lines[J]. The Cartographic Journal，1995，32(2)：121-125.

[7] DOUGLAS D H，PEUCKER T K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature [J]. Cartographica：the International Journal for Geographic Information and Geovisualization，1973，10(2)：112-122.

[8] VISVALINGAM M，WHYATT J D. Line generalisation by repeated elimination of points[J]. The Cartographic Journal，1993，30(1)：46-51.

[9] LI Z，郭庆胜. 基于客观综合自然规律的线状要素自动综合的算法[J]. 武测译文，1994(1)：49-58.

[10] 朱长青，王玉海，李清泉，等. 基于小波分析的等高线数据压缩模型[J]. 中国图象图形学报：A 辑，2004，9(7)：841-845.

[11] WANG B，SHU H，LUO L. A genetic algorithm with chromosome-repairing for min-# and min-ε polygonal approximation of digital curves[J]. Journal of Visual Communication and Image Representation，2009，20(1)：45-56.

[12] WANG Bin，BROWN D，ZHANG Xiaozheng，et al. Polygonal approximation using integer particle swarm optimization [J]. Information Sciences，2014，278：311-326.

[13] 刘生建，杨艳，周永权. 一种群体智能算法—狮群算法[J]. 模式识别与人工智能，2018，31(5)：431-441.

[14] ZHU Xiaobo，ZHOU Tinggang，ZENG Bo. A parallel compression algorithm for multilevel river linear vector data considering spatial adjacency relations [J]. Journal of Southwest (Natural Science Edition)，2017，39(2)：100-106.

[15] LIU Minshi，YI Long，FEI Lifan. Line simplification of three-dimensional drainage considering topological consistency[J]. Acta Geodaetica et Cartographica Sinica，2016，45(4)：494.

[16] 费若男，孔政，宫振峰，等. 基于改进的道格拉斯-普克算法确定大气激光雷达消光系数边界值[J]. 中国激光，2023，50(14)：202-214.

[17] 陈信强，徐祥龙，彭静，等. 基于 Douglas-Peucker 和 Quick Bundles 算法的水上交通模式识别[J]. 上海海事大学学报，2022，43(3)：1-6.

[18] 何宽，孙瑞，官云兰，等. 基于逐点前进法的点云数据精简[J]. 测绘通报，2022(9)：167-169.

[19] 陈万利，张梅，冯涛. 基于 Douglas-Peucker 融合闵式距离的锂电池健康因子提取及 SOH 预测[J]. 储能科学与技术，2022，11(10)：3306-3315.

[20] 王荣，闫浩文，禄小敏. Douglas-Peucker 算法全自动化的多尺度空间相似关系方法[J]. 地球信息科学学报，2021，23(10)：1767-1777.

[21] 周腾，李晓妍. 改进的道格拉斯-普克算法在电信代维系统的应用[J]. 计算机与数字工程，2020，48(7)：1576-1579，1692.

[22] 杨伟. GIS 应用中的矢量数据压缩算法研究[D]. 成都：四川师范大学，2017.

[23] TANG Chunhua，WANG Han，ZHAO Jiahuan，et al. A method for compressing AIS trajectory data based on the adaptive-threshold Douglas-Peucker algorithm [J]. Ocean Engineering，2021，232：109041.

[24] ZHAO Lianbin，SHI Guoyou. A trajectory clustering method based on Douglas-Peucker compression and density for marine traffic pattern recognition [J]. Ocean Engineering，2019，172：456-467.