

面向大规模地理矢量线数据的多层级实时可视化技术*

刘泽邦, 陈 犇, 马梦宇, 杨岸然, 钟志农, 景 宁
(国防科技大学 电子科学学院, 湖南 长沙 410073)

摘要:针对主流方法难以满足多层级实时可视化的需求,提出面向大规模地理矢量线数据的多层级实时可视化技术。建立面向多层级瓦片绘制的自适应可视化模型,设计像元四叉 R(pixel quad R, PQR)树空间索引和基于PQR树的自适应可视化算法,分别用于支撑模型的数据组织和可视绘制。在10亿规模数据集上的实验表明:该技术在0.57 s内可计算任一层级上的可视结果,并且计算耗时大幅小于主流方法。当数据规模急剧增长时,该技术在各显示层级上仍具有较好的可视性能,最低可视速率超过100张/s,大幅优于主流方法。该技术在单机条件下即可支撑大规模地理矢量线数据的多层级实时可视化,在空间大数据探索分析领域具备较好的应用前景。

关键词:大数据;地理矢量线数据;空间索引;多层级;实时可视化

中图分类号:P208 文献标志码:A 开放科学(资源服务)标识码(OSID):

文章编号:1001-2486(2023)05-173-11



听语音
与作者互动
聊科研

Multilevel real-time visualization technology for large-scale geographic vector linestring data

LIU Zebang, CHEN Luo, MA Mengyu, YANG Anran, ZHONG Zhinong, JING Ning

(College of Electronic Science and Technology, National University of Defense Technology, Changsha 410073, China)

Abstract: Aiming at the difficulty of mainstream methods to support the multilevel real-time visualization of large-scale geographic vector linestring data, a multilevel real-time visualization technique for large-scale geographic vector linestring data was proposed. An adaptive visualization model for multilevel tile rendering was established, and a PQR (pixel-quad-R) tree spatial index and an adaptive visualization algorithm based on PQR-tree were designed to support the data organization and visualization of the model, respectively. Experiments on billion-scale datasets show that the technique can calculate visualization results at any zoom level within 0.57 s. Meanwhile, its visualization time is significantly less than mainstream methods. When the data scale increases sharply, the technology still has good visualization performance at each zoom level, and the lowest visualization rate exceeds 100 tiles/s, which is much better than mainstream methods. The technique can support multilevel real-time visualization of large-scale geographic vector linestring data in the single machine, and has a good application prospect in the field of exploration and analysis of spatial big data.

Keywords: big data; geographic vector linestring data; spatial index; multilevel; real-time visualization

地理矢量数据模型通常以点、线、面等离散对象来描述地理表面空间要素的形状和位置,以及刻画地理空间现象的范围与分布,是全空间地理信息系统描述时空实体的核心数据模型^[1]。地理矢量线数据作为地理矢量数据的重要组成部分,通过一系列顺序排列的点来实现对现实世界的表达,被广泛用于道路、行政区划、河流水系、建筑物轮廓等多种线状地物的描述和表达。随着地理空间数据采集技术的快速发展,地理矢量线数据的规模呈现爆炸性增长态势。根据2021年交

通运输行业发展统计公报数据显示,截至2021年底全国公路总里程已达 5.2807×10^7 km,涉及的道路矢量数量规模达到了千万级^[2];截至2022年,OSM(OpenStreetMap)中的地理矢量线要素的规模已达到亿级^[3]。这些体量庞大且不断增长的地理矢量线数据不仅数据量巨大,而且绝大部分的数据仅仅以表格或文本形式呈现,用户需耗费大量的时间和精力进行数据的处理和分析。

空间数据可视化作为空间分析的重要手段,是探索挖掘空间数据的基础^[4-5],有助于用户直

* 收稿日期:2023-03-20

基金项目:国家自然科学基金资助项目(42101432,41971362);湖南省自然科学基金资助项目(2022JJ40546)

作者简介:刘泽邦(1998—),男,湖南衡阳人,博士研究生,E-mail:liuzebang19@nudt.edu.cn;

陈犇(通信作者),男,上海人,教授,博士,博士生导师,E-mail:luochen@nudt.edu.cn

观地对空间数据进行总结、分析和推理^[6-7]。随着网络信息技术的快速发展,基于瓦片金字塔技术^[8]的网络瓦片地图服务被提出并用于在互联网浏览地理矢量数据,大规模地理矢量数据的高效可视化成为当前地理信息科学领域的研究热点^[9-13]。

在可视过程中,需要遵循经典的可视分析探索流程^[14]:既能以鹰眼视角浏览数据的总体分布,又可进行缩放浏览数据的个体细节。为提升可视化效率,单机上的可视化方法通过数据采样、聚合、简化等一系列减少数据规模的方法来实现快速可视化响应^[15-20]。一些研究利用 GPU 的图形处理优势来加速地理矢量数据可视化处理过程^[21-23]。矢量瓦片技术作为近期流行的可视化技术^[24-25],可支持在客户端直接对数据进行样式渲染。文献[26]实现了大规模卫星点数据的多分辨率的热区分布图展示。Eldawy 等^[27]实现了散点图、路网图、热力图等多种可视化效果。YU 等^[28-29]实现了亿级规模空间数据的地图可视化。文献[30-31]采用预先生成部分可视化结果的思想实现了高分辨率的地图实时可视化。然而,上述可视化方法采用数据导向的计算方式,以单个矢量要素作为计算单元进行可视化计算。随着数据规模的急剧增长,会产生地图瓦片数据集生成耗时长,规模大,样式难以交互更改等问题,即使采用高性能并行计算技术也难以满足无缝数据浏览、高效实时渲染等可视化需求^[32]。

1 可视化模型描述

当前主流可视化方法以数据为计算导向,如图 1 所示,其核心思想是以单个矢量要素作为计算单元,将其转换为便于屏幕显示的栅格对象,最后合并得到可视化结果。

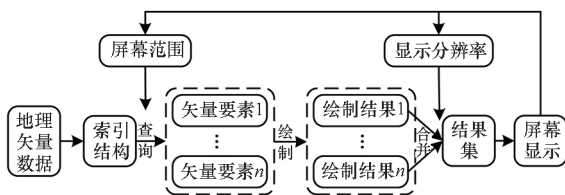


图 1 传统数据导向的可视化模型

Fig. 1 Traditional data-driven visualization model

显示导向的可视化模型^[33-36]如图 2 所示。该模型以屏幕像元为计算单元,将依赖矢量要素规模的计算复杂度转换为依赖像元规模的计算复杂度,大幅提升了计算效率。

在可视浏览时,可视瓦片的属性不尽相同,如

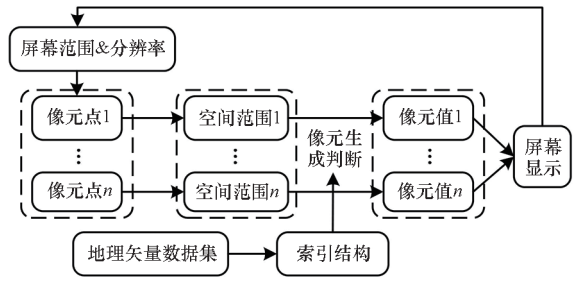


图 2 显示导向的可视化模型

Fig. 2 Display-driven visualization model

显示层级或高或低,要素数量或多或少,导致绘制效率差异较大。为实现各显示层级上的实时可视化,提出面向多层级瓦片绘制的自适应可视化模型。如图 3 所示,模型对瓦片绘制任务进行自适应判定,选择最优方式绘制可视结果。当采用显示导向可视化算法,进行像元生成判定;当采用数据导向可视化算法,检索瓦片内矢量要素进行栅格化。在模型中,地理矢量数据需转换成特定的空间索引用以支撑多层级瓦片绘制,该索引一方面能为瓦片绘制算法的自适应选取提供支撑,另一方面能兼顾显示导向与数据导向两种可视化算法的计算需求。最后,在自适应判定器中设定判定因子,根据瓦片绘制任务的判定结果来选取可视化算法。判定因子包括:瓦片显示层级,即显示层级越低,则线要素越多,可视化耗时长;瓦片内矢量线要素数量,即线要素越多,可视化耗时越长;瓦片内矢量线要素总长度,即线要素总长度越长,可视化耗时越长。

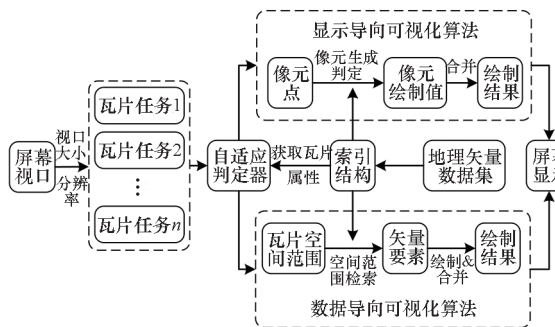


图 3 面向多层级瓦片绘制的自适应可视化模型

Fig. 3 Adaptive visualization model for multilevel tile drawing

2 像元四叉 R 树索引

2.1 索引结构设计

在面向多层级瓦片绘制的自适应可视化模型中,为绘制任意显示层级上的一张瓦片图,数据导向的可视化方法需快速检索与该瓦片对应空间范

围拓扑相交的矢量线要素;显示导向的可视化方法需要判断像元对应空间范围是否与任一矢量线要素拓扑相交。

为实现快速可视化所需的拓扑相交和数据检索的计算需求,设计了一种PQR树空间索引。如图4所示,PQR树是由多个索引结点连接的类四叉树结构,有别于常规四叉树以数据集最小外包框作为根结点索引范围,PQR树将全球空间范围

作为根结点索引范围,再不断向下进行递归四分得到索引结点,索引结点共五种类型:根结点、左上结点、右上结点、左下结点、右下结点。在索引结点中,记录结点的空间范围、地理编码、结点属性与结点指针等信息。当达到预先设定的最高层级后则不再进行划分,在PQR树的最高层级上,每个索引结点连接一个R树,对与索引结点索引范围拓扑相交的矢量线要素进行高效组织。

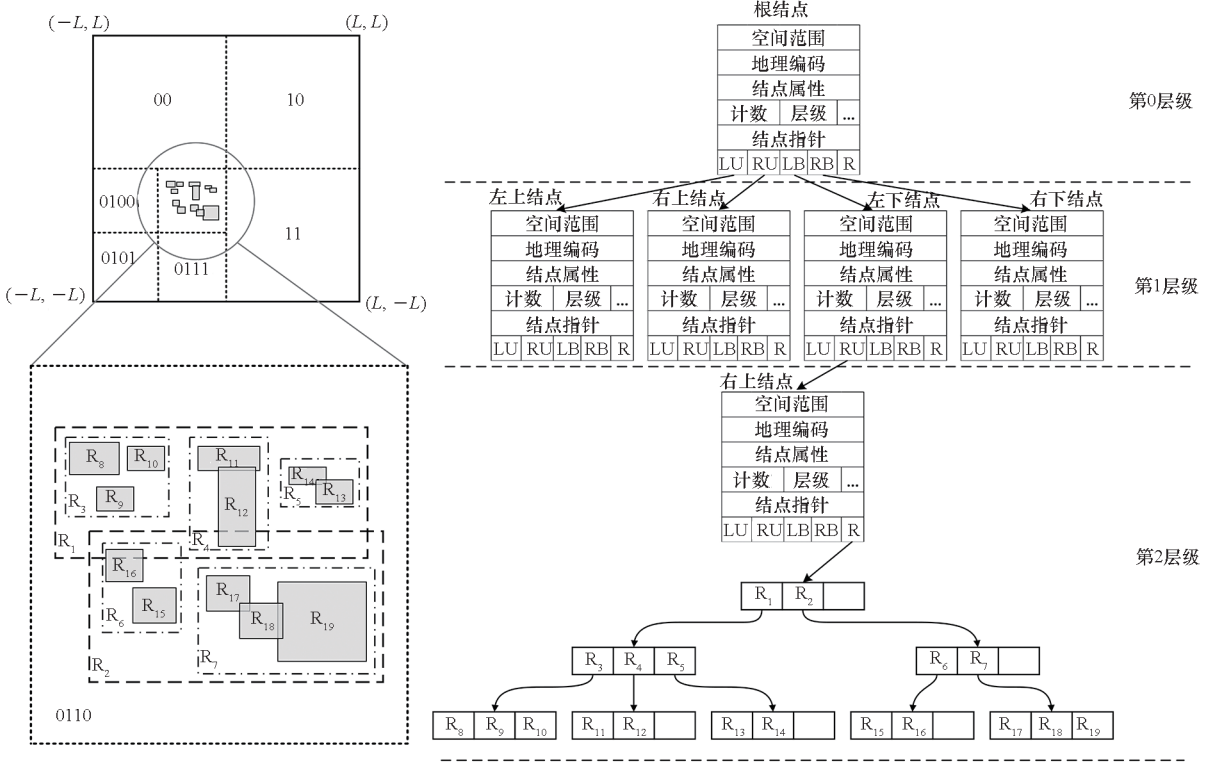


图4 PQR树索引结构设计

Fig. 4 PQR-tree index layout

PQR树索引结点说明如下:①基于Geohash的编码方法来获取索引结点编码。如图4所示,从根结点向下编码,每向下递归划分一层:左上、右上、左下、右下结点的编码分别为“00”,“10”,“01”和“11”,不同层级结点的编码长度不一致,最终得到二进制形式的编码作为索引结点编码。②对于矢量线数据,选取要素数量 *count*、显示层级 *level*、要素长度 *length* 三个判定因子作为结点属性。③结点指针包括四个子结点指针 LU-*ptr*、RU-*ptr*、LB-*ptr*、RB-*ptr* 和一个R树指针 R-*ptr*。非叶子结点的四个子结点指针分别指向该结点索引范围四分后的四个子区域所对应的子索引结点,R树指针为空;叶子结点的四个子结点指针均为空,R树指针指向该索引结点所连接的R树。

PQR树的索引结点的索引范围与瓦片/像素对应空间范围是唯一映射的,这一特性使其可同

时满足自适应判定、数据导向与显示导向可视化方法的计算需求:①只需快速找到瓦片对应的索引结点,并对其中记录的判定因子进行比较,即可完成自适应判定;②对于瓦片对应的索引结点,对该结点连接的R树执行空间检索获取瓦片对应空间范围内的矢量要素,即可完成数据导向可视化方法的计算需求;③对于瓦片对应的索引结点,只需从该结点向下判断像素对应的索引结点是否存在,即可完成显示导向可视化方法的计算需求。

2.2 索引构建

地理矢量线数据的PQR树索引构建算法的核心思想是从根结点“自顶而下”地将矢量线要素与全球地理空间范围递归四分的结果进行拓扑相交判断(“位于”和“叠置”)来生成索引结点,在达到最高层级后,将矢量线要素组织生成R树结构。由于像元对应空间范围是由瓦片对应空间

范围递归划分八次得到,为满足显示导向可视化方法的计算需求,PQR 树的层级不低于 8。

在创建索引结点时,最常采用近似表达的方式将矢量线要素的最小外包框 (minimum boundary box, MBB) 与空间范围框进行判定。考虑当在低显示层级时,矢量线要素完全位于空间范围框内,而在高显示层级时,矢量要素只与部分空间范围框相交。为提高计算效率,将线要素表示为双层外包框结构,如图 5 所示,其包含了线要素的 MBB 和线要素中每条线段的 MBB。在相交判断时,依次采用线要素的 MBB 和每条子段的 MBB 用于计算,两层过滤后再判断每条子段与空间范围框是否空间拓扑相交。

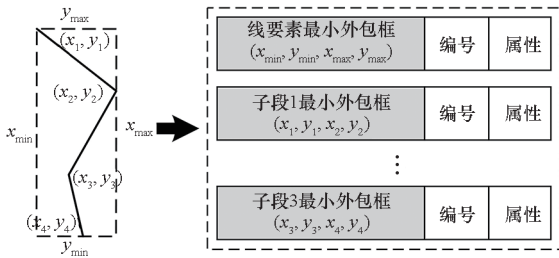


图 5 地理矢量线要素的双层级最小外包框结构
Fig. 5 Double MBB structure of geographic vector linestring features

PQR 树索引构建流程包括:①创建根结点,对于任一地理矢量数据集,根结点的空间范围均为全球地理空间范围。②逐一遍历数据集中地理矢量线要素,将其表示为双层级外包框结构。③将根结点的空间范围划分为四个子空间,分别与线要素的 MBB 进行空间关系判定,若线要素位于某个子空间内,则创建相应的索引结点,并从该索引结点继续向下进行递归判定,直至生成包含要素 MBB 的最小索引结点。④依次选取要素的每条子段的 MBB,将其分别与最小索引结点的空间范围划分后所得的四个子空间进行空间关系判定。⑤若子段的 MBB 位于某个子空间内,则创建相应的索引结点;若子段的 MBB 与某些子空间相叠置,则再精细判定子段与这些子空间是否相叠置来创建相应的索引结点;继续从新创建的索引结点向下递归判定,直至设定的最高层级。⑥分别将要素子段插入与其相交的索引结点所连接的 R 树中。⑦当所有的矢量线要素判定完毕后,构建得到最终的 PQR 树索引。

3 基于像元四叉 R 树的自适应可视化

在面向多层次瓦片绘制的自适应可视化模型

中,为快速绘制各个显示层级上的瓦片图,设计基于 PQR 树的自适应可视化 (PQR-tree-based adaptive visualization, PAV) 算法。PAV 对判定因子的判定流程进行了设定:当采用显示导向可视化方法,核心是判定像元对应的索引结点在 PQR 树中是否存在;当采用数据导向可视化方法,核心是从 PQR 树中快速检索瓦片对应空间范围内的矢量线要素。

根据 2.2 节,对线数据集构建 PQR 树后,与线要素存在空间拓扑相交的空间范围框会生成对应的索引结点。所以对于待绘制的瓦片任务,需要判断瓦片对应的索引结点在 PQR 树中是否存在来决定瓦片是否需要被绘制。同时,在显示导向的可视化绘制时,也需要判断像元对应的索引结点在 PQR 树中是否存在来决定像元值是否需要被生成。基于此,设计了索引结点查找函数 SEARCHNODE,如算法 1 所示,“自顶而下”地从设定索引结点向下查找对应的索引结点。函数输入包括开始查询的索引结点 s_{Node} 和瓦片/像元对应空间范围框的地理编码 r_{Code} ,当查询到对应的索引结点时进行输出,反之输出为空。算法主要步骤为:①逐个遍历初始结点的子结点指针,判断子结点 c_{Node} 是否存在;②当子结点存在时,获取 c_{Node} 的地理编码 c_{Code} ;③判断 c_{Code} 与 r_{Code} 长度是否相等;若两者不等,取两者的前缀对应位数进行异或操作,称为前缀匹配;④若操作结果为 0,将

算法 1 索引结点查找函数

Alg. 1 Searching function of index node

输入:查询的索引结点 s_{Node} , Geohash 编码 r_{Code}
输出:查询到的索引结点 c_{Node}

```

1. Function SEARCHNODE( $s_{Node}, r_{Code}$ )
2.   for  $s_{Node}$  的子结点  $c_{Node}$  do
3.     if  $c_{Node}$  不为空 then
4.       获取  $c_{Node}$  的地理编码  $c_{Code}$ 
5.       if  $r_{Code}$  与  $c_{Code}$  长度不一致 then
6.         将  $r_{Code}$  与  $c_{Code}$  进行前缀匹配
7.         if 匹配结果为 0 then
8.           SEARCHNODE( $c_{Node}, r_{Code}$ )
9.       end if
10.      else if  $r_{Code} == c_{Code}$  then
11.        return  $c_{Node}$ 
12.      end if
13.    end if
14.  end for
15. return null

```


c_{Node} 与 r_{Code} 作为函数输入,递归执行索引结点生成函数;⑤直至当 r_{Code} 与 c_{Code} 完全相等,输出 c_{Node} 为与空间范围框相对应的索引结点,其余情况下均输出为空。

当瓦片对应的索引结点存在时,则说明瓦片需要被绘制。此时,PAV获取到该瓦片对应的索引结点中的属性信息(显示层级 $level$ 、线要素数量 $count$ 、线要素总长度 $length$),并与三个判定因子的阈值进行比较,具体的判定流程如下:①判断层级信息 $level$ 与 δ_{level} 的大小,当 $level$ 大于 δ_{level} 时,则选择数据导向的可视化方法;反之继续判断。②判断 $count$ 与 δ_{count} 的大小,当 $count$ 大于 δ_{count} 时,选择显示导向的可视化方法;反之继续判断。③判断 $length$ 与 δ_{length} 的大小,当 $length$ 大于 δ_{length} 时,选择显示导向的像素生成策略;反之选择数据导向的可视化方法。

PAV的具体实现主要分为五步:①调用算法1的索引结点查找函数,判断待绘制瓦片对应的索引结点在索引中是否存在,若不存在则直接输出空白的瓦片图;反之继续执行。②获取索引结点的结点属性信息($level, count, length$),与设定的判定因子阈值进行判定,自适应地选取可视化方法。③若选取显示导向的可视化方法,则遍历瓦片的每个像元,继续调用算法1的索引结点查找函数,判断像元对应的索引结点是否存在,若存在则直接在瓦片图上生成像元值。④若选取数据导向的可视化方法,则对该索引结点连接的R树执行空间范围检索操作,得到地理矢量线要素,并逐一进行栅格化处理生成像元值。⑤最后输出得到瓦片图。

PAV的优势在于:采用显示导向的可视化方法时,查找像素对应的索引结点是否存在只有唯一的判断路径,采用简单的二进制运算即可快速生成瓦片图;采用数据导向的可视化方法时,在查找瓦片对应的索引结点是否存在同样只有唯一的判断路径,进而快速检索到矢量要素并进行栅格化生成瓦片图。最终可实现各个显示层级上瓦片图的快速绘制。

4 实验与分析

实验数据集如表1所示,来自开源志愿者网络地图服务平台OSM。从 L_1 至 L_7 ,数据规模逐渐增大。 L_7 包含OSM全球所有线要素数据,其中线要素的规模达到了10亿级别。实验环境:内存为256 GB,核数为32的2.6 GHz的Inter处理器,且处理系统为Ubuntu 20.04。

表1 实验数据集

Tab. 1 Datasets used in the experiments

数据集	大小	要素数量	数据规模
L_1 :OSM 瑞士道路	135.2 MB	597 829	6 991 831
L_2 :OSM 成都市道路	412.6 MB	1 119 753	20 748 198
L_3 :OSM 中国道路	1.2 GB	4 550 478	54 847 749
L_4 :OSM 全球水域边界	6.7 GB	9 211 000	376 208 235
L_5 :OSM 全球绿地边界	7.2 GB	11 634 551	396 527 008
L_6 :OSM 全球道路	16.7 GB	72 336 396	717 048 198
L_7 :OSM 全球线要素	32.9 GB	106 268 554	1 573 469 984

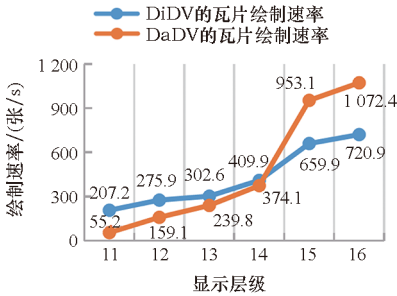
为确立PAV中判定因子的阈值,实验一选取 $L_1 \sim L_6$ 作为测试数据集,对比显示导向的可视化方法(display-driven visualization, DiDV)和数据导向的可视化方法(data-driven visualization, DaDV)在不同的判定因子取值下的瓦片绘制性能,为PAV中的自适应判定提供依据。实验二在所有线数据集上分析PAV算法的可视化性能,其中 L_7 作为测试集用以验证算法的泛化能力,最终验证PAV算法可以支撑大规模地理矢量数据集的多层级实时交互式可视化。实验三将PAV与当前主流可视化方法进行比较,验证PAV算法的优势。在实验中所有数据集均被预先转换成PQR索引以支撑PAV算法。同时,由于每个瓦片的绘制任务间相互独立,故采用多进程消息传递框架(message passing interface, MPI)与OpenMp混合并行的方式来提升瓦片绘制的性能,在所有实验中均启动了8个MPI进程(每个进程中包含4个OpenMp线程)用于绘制瓦片。

4.1 实验一:判定因子阈值设定

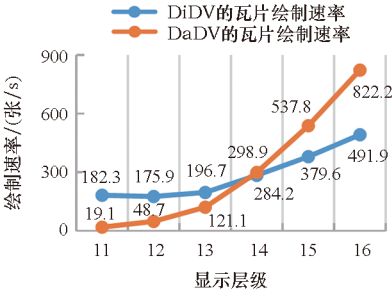
本实验用于获取判定因子的阈值。采用DiDV与DaDV绘制瓦片图,统计每张瓦片判定因子的值,评估瓦片的显示层级、线要素数量、线要素总长度对两种方法性能的影响。

为分析显示层级的影响,在各显示层级随机选取2 000张非空瓦片,计算两种方法在各显示层级上的绘制速率,实验结果如图6所示,当显示

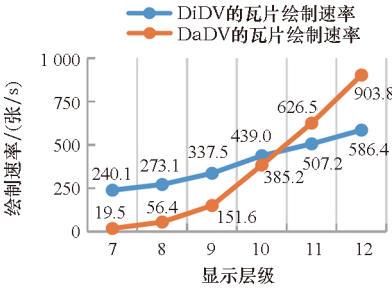
层级较低时, DiDV 的可视化性能明显优于 DaDV, 并且当数据规模较大时的性能优势更明显, 在 L_4 数据集上当显示层级为 6 时, DiDV 的绘制速率为 DaDV 的 35 倍; 随着显示层级升高, DaDV 的效率显著提升并开始超越 DiDV, 在 L_6 数据集上的最大绘制速率达到 563.8 张/s。综上可知, 当显示层级较低时应选择 DiDV, 反之选择 DaDV。在设定阈值时, 将完全包含数据集的最小瓦片层级作为其空间尺度值, 例如全球空间的数据集的空间尺度则为 0。当数据集的空间尺度不同时, 显示层级阈值 δ_{level} 也不同, 对于 $L_1 \sim L_6$, δ_{level}



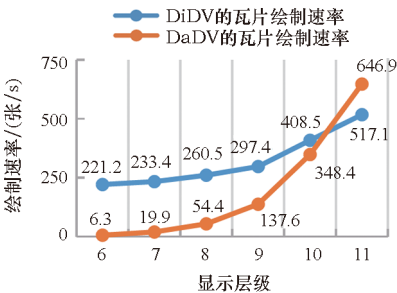
(a) L_1



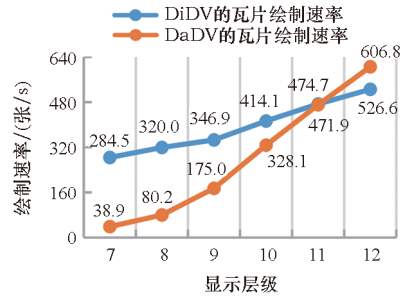
(b) L_2



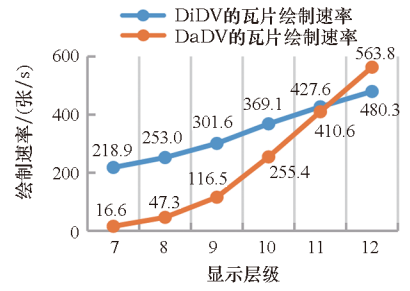
(c) L_3



(d) L_4



(e) L_5

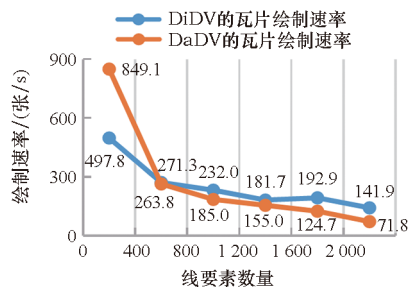


(f) L_6

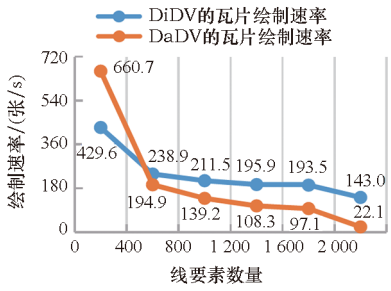
图 6 瓦片绘制速率随显示层级变化的对比
 Fig. 6 Comparison of tile drawing speed at different zoom levels

的值比空间尺度的值大 8 至 11, 为满足所有数据集在各个显示层级上的快速可视化, 以数据集的空间尺度值 + 11 后的值设为 PAV 算法中的 δ_{level} 。

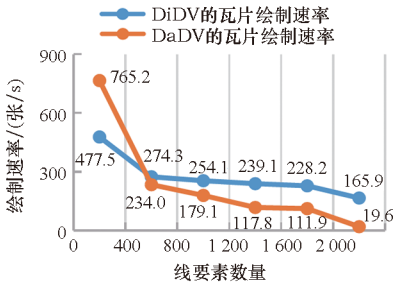
为分析瓦片空间范围内线要素数量的影响, 设定六个不同的范围区间: 0 ~ 400、400 ~ 800、800 ~ 1 200、1 200 ~ 1 600、1 600 ~ 2 000、> 2 000, 对于每个范围区间随机选取 2 000 张非空瓦片, 计算 DiDV 和 DaDV 的瓦片绘制速率, 实验结果如图 7 所示。DaDV 在线要素数量较小时表现出优于 DiDV 的可视化性能, 然而其性能随线要素数量的增长而显著下降, 而 DiDV 的瓦片绘制速率维持着相对稳定的水平。对于数据集 $L_1 \sim L_6$, 实验结果显示在线要素数为 400 ~ 800 范围内 DiDV 的瓦片绘制速率开始超越 DaDV, 由此以 800 作为 PAV 算法中的线要素数量阈值 δ_{count} 。



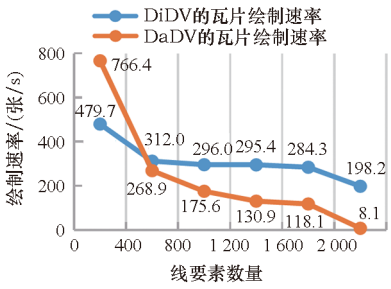
(a) L_1



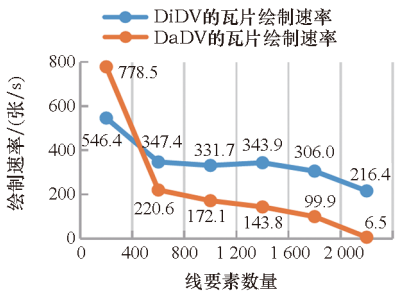
(b) L_2



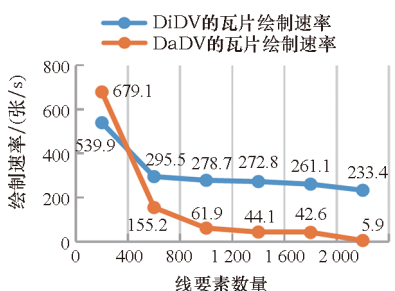
(c) L_3



(d) L_4

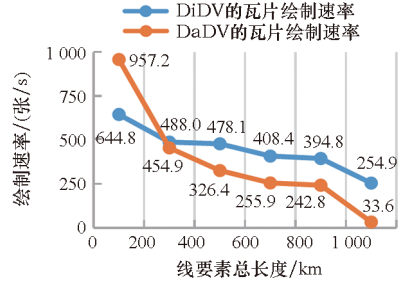


(e) L_5

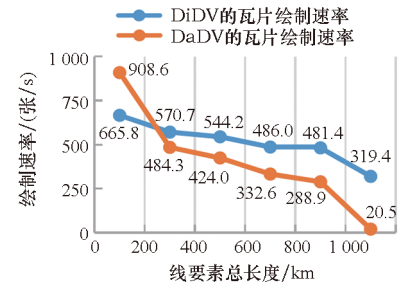


(f) L_6

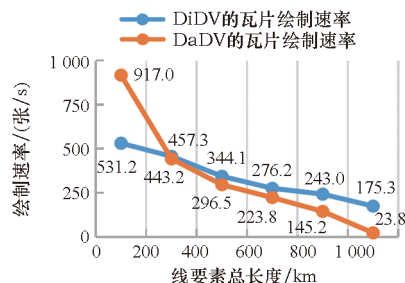
为分析瓦片空间范围内线要素的影响,设定六个不同的范围区间:0 ~ 200 km、200 ~ 400 km、400 ~ 600 km、600 ~ 800 km、800 ~ 1 000 km、> 1 000 km。对于每个范围区间随机选取 2 000 张非空瓦片,计算 DiDV 和 DaDV 的瓦片绘制速率,实验结果如图 8 所示, DaDV 在线要素总长度较小时表现出优于 DiDV 的可视化性能,然而其性能随线要素总长度的增长而显著下降,而 DiDV 的瓦片绘制速率维持着相对稳定的水平。对于数据集 $L_1 \sim L_6$,实验结果显示在线要素总长度为 200 ~ 400 km 的范围区间上 DiDV 的瓦片绘制速率开始超越 DaDV,由此以 400 km 作为 PAV 算法



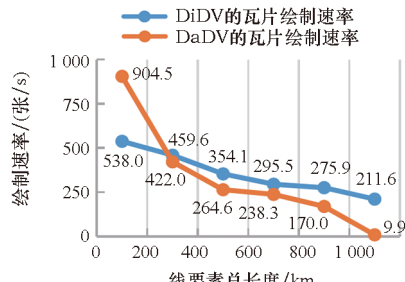
(a) L_1



(b) L_2



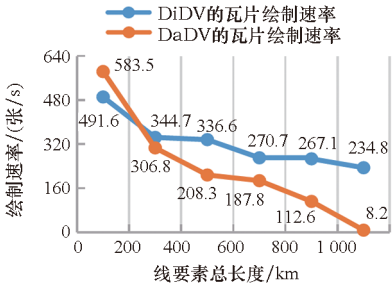
(c) L_3



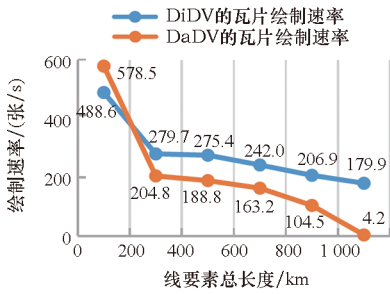
(d) L_4

图7 瓦片绘制速率随线要素数量变化的对比

Fig. 7 Comparison of tile drawing speed varies with the number of linestring features



(e) L_5



(f) L_6

图 8 瓦片绘制速率随线要素总长度变化的对比

Fig. 8 Comparison of tile drawing speed varies with the total length of linestring features

中的线要素总长度阈值 δ_{length} 。

4.2 实验二:支撑多层级实时可视化性能

为验证 PAV 算法具备支撑大规模地理矢量线数据的多层级实时可视化的能力,将实验一确定的阈值 ($\delta_{level} = 11$ 、 $\delta_{count} = 800$ 、 $\delta_{length} = 400$ km) 代入 PAV 算法中,对于所有数据集 $L_1 \sim L_7$,在 0~20 显示层级上随机选择 5 000 张非空瓦片,分别统计生成 5 000 张瓦片的总耗时和生成每张可视化瓦片的绘制时间。实验结果见图 9,对于所有数据集,PAV 算法均维持着相对稳定高效的可视化性能,瓦片绘制速率最低也能达到 366.5 张/s,远高于实时浏览数据所需的瓦片请求速率(通常不超过 100 张/s)。在 10 亿规模测试数据集 L_7 上,瓦片生成总耗时仅为 13.6 s,瓦片绘制速率达到 366.5 张/s。同时,从 $L_1 \sim L_7$ 数据集规模显著增长,而算法的瓦片生成总耗时并无明显的增长,且瓦片绘制速率并无显著的下降趋势,说明其具有对数据集规模不敏感的特性,非常适用于对大规模地理矢量线数据的多层级实时可视化。图 10 为生成 5 000 张瓦片的时间分布箱型图,其中“●”表示瓦片的平均生成时间,从结果可知,对于所有数据集,绝大部分的瓦片在 30 ms 内可完成绘制,在 10 亿规模测试数据集 L_7 上,瓦片的最长生成时间仅为 44 ms。假设当前浏览器发送了 100 个瓦片请求,所有任务通过 13 轮 ($100 \div 8$) 绘

制,最差情况也能在 0.57 s ((13×44) ms) 内完成任务,完全可以满足实时可视化的需求。综上,结合图 9 和图 10 结果分析可知,PAV 算法可以高效地支撑 10 亿级规模的地理矢量线数据的多层级实时可视化。

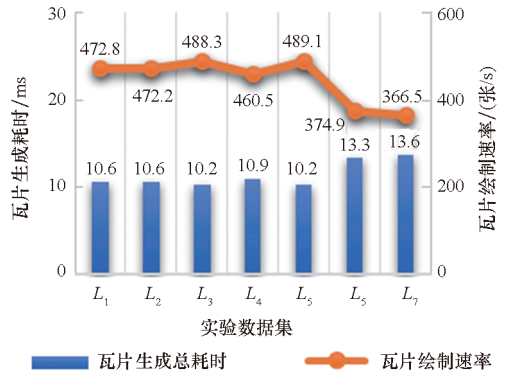


图 9 生成 5 000 张瓦片的总耗时和绘制速率

Fig. 9 Total time and tile drawing speed of drawing 5 000 tiles

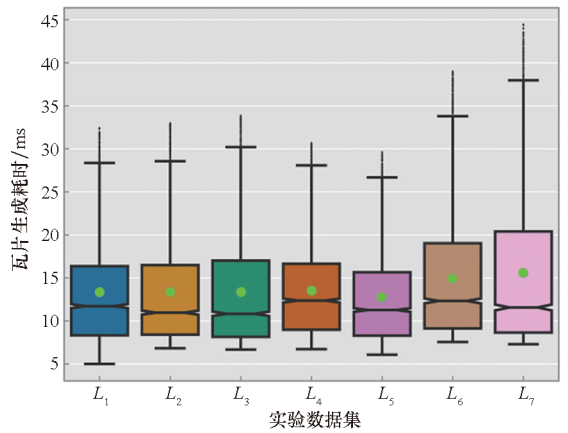


图 10 生成 5 000 张瓦片的时间分布箱型图

Fig. 10 Box diagram of time distribution of drawing 5 000 tiles

4.3 实验三:与主流可视化方法的性能对比

为验证 PAV 算法的优势,将其与当前两个主流的空间数据可视化方法 GeoSparkViz 和 HadoopViz 进行比较,对于所有数据集,分别采用 3 种方法生成 0~8 显示层级上的所有可视化瓦片,得到三种方法的总耗时对比,如图 11 所示,对于所有数据集,GeoSparkViz 生成瓦片的耗时少于 HadoopViz,GeoSparkViz 的瓦片生成性能相较 HadoopViz 更优,然而随着数据规模增长,这两种方法的瓦片生成性能急剧下降。而 PAV 生成可视化瓦片的耗时均远少于 GeoSparkViz 与 HadoopViz,表现出优异的可视化性能,对于体量超过亿级的数据集 $L_4 \sim L_7$,PAV 的瓦片生成耗时

分别仅为 GeoSparkViz 的 13.2%、11.6%、8.1%、7.5%，且数据规模越大，PAV 相较于 GeoSparkViz 的瓦片生成性能优势越明显。从 $L_1 \sim L_7$ ，随着数据规模的增长，PAV 的耗时并未出现大幅增长，验证了 PAV 具有对数据规模不敏感的特性，能够支撑大规模地理矢量线数据的多层级实时可视化。

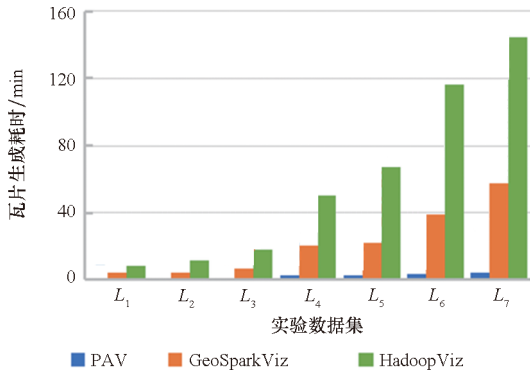
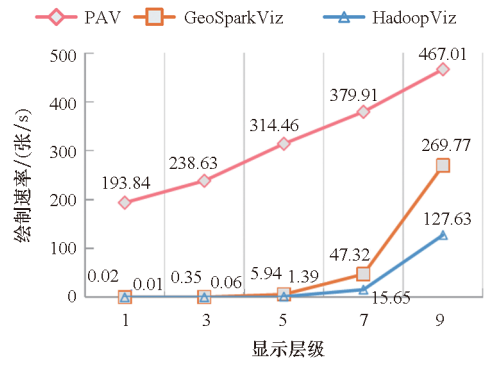


图 11 生成可视化瓦片的总耗时

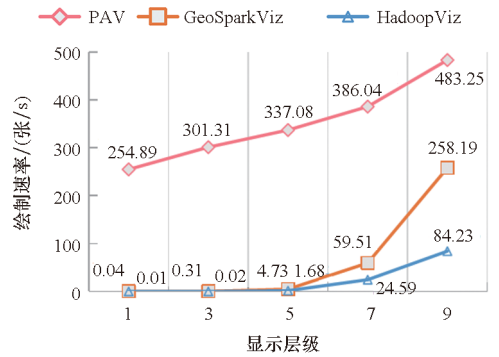
Fig. 11 Total time to render all tiles in zoom levels 0 ~ 8

为进一步对比三种方法在不同显示层级上的可视化性能，图 12 对比了三种方法在亿级规模数据集 $L_4 \sim L_7$ 上第 1、3、5、7、9 显示层级上的瓦片绘制速率，从实验结果可知：GeoSparkViz 和 HadoopViz 在较高的显示层级上具有较高的瓦片绘制速率，其中 GeoSparkViz 的瓦片绘制速率优于 HadoopViz，GeoSparkViz 在 L_4 数据集的第 9 显示层级上的瓦片绘制速率能达到 269.77 张/s。然而，随着显示层级的降低，GeoSparkViz 与 HadoopViz 的瓦片绘制速率大幅下降，按照可视化浏览从整体到精细的原则，GeoSparkViz 与 HadoopViz 无法实现大规模地理矢量线数据的多层级实时可视化。而 PAV 在所有数据集的各个显示层级上均具有较高的瓦片绘制速率，特别当显示层级较低时，其瓦片绘制速率最差在 L_7 数据集上也能达到 115.29 张/s，大幅优于 GeoSparkViz (0.01 张/s) 与 HadoopViz (0.001 张/s) 的可视化性能。

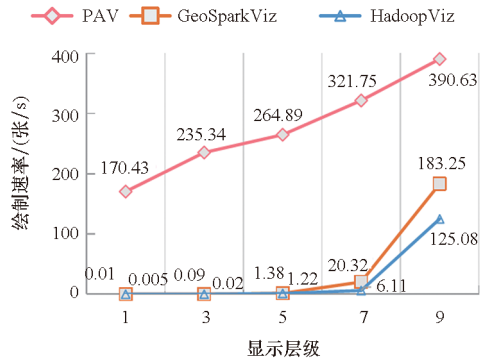
从上述结果可知：对于反映可视化效率的两个主要指标瓦片生成总耗时和各显示层级的瓦片绘制速率，PAV 均大幅优于主流可视化方法 GeoSparkViz 和 HadoopViz。原因在于：GeoSparkViz 和 HadoopViz 采用数据导向的计算方式，从 L_1 至 L_7 ，随着数据规模的增长，这两种方法的瓦片生成总时间必然会大幅增长；并且对于同一数据集，当在较高的显示层级上，瓦片内的矢量线要素的规模较小，所以 GeoSparkViz 和



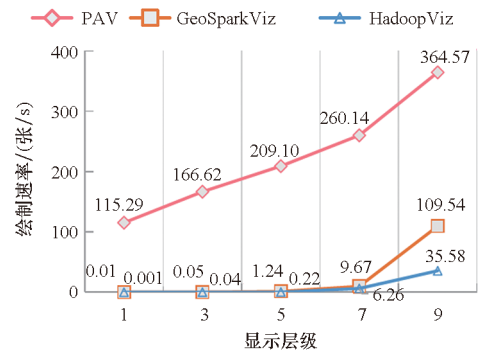
(a) L_4



(b) L_5



(c) L_6



(d) L_7

图 12 各显示层级上的瓦片生成速率对比
Fig. 12 Comparison of tile drawing speed on each zoom level

HadoopViz 具备相对较高的瓦片绘制速率，随着显示层级的降低，数据规模急剧增长，某张瓦片内

的矢量线要素可能达到千万甚至上亿量级,从而使得 GeoSparkViz 和 HadoopViz 在低显示层级上的瓦片绘制速率显著下降。PAV 采用显示导向与数据导向相结合的自适应可视化方法,当数据规模较小或显示层级较高时使用数据导向的可视化方法能快速绘制瓦片,反之采用显示导向的可视化方法直接计算像元点的方式能大幅提升可视化效率,由此在较低显示层级上仍具有较高的瓦片绘制速率,所以 PAV 在所有数据集的各个显示层级上均具有较高的瓦片绘制速率,并且总的瓦片生成总耗时大幅降低。综上所述,PAV 相较现有主流的数据导向的可视化方法 GeoSparkViz 与 HadoopViz 具有明显优势,能够支撑大规模地理矢量线数据的多层级实时可视化。

5 结论

针对大规模地理矢量数据可视化这一当前 GIS 领域研究的热点问题,以大规模地理矢量线数据为研究对象提出高效的多层级可视化技术。该技术采用瓦片金字塔模型来展示地理矢量线数据,以实现高效的瓦片绘制性能为计算目标,提出面向多层级瓦片绘制的自适应可视化模型,在模型中通过自适应判定来选择最优的显示导向或数据导向的可视化方法,同时设计空间索引结构和可视化算法来支撑模型中涉及的数据组织和可视化流程。实验证明该技术在各个显示层级上均具有高效的瓦片绘制性能,可在单机条件下支撑亿级规模地理矢量线数据的多层级实时可视化,在空间数据的可视化探索上具有较好的应用前景。

参考文献 (References)

- [1] 周成虎. 全空间地理信息系统展望[J]. 地理科学进展, 2015, 34(2): 129 - 131.
ZHOU C H. Prospects on pan-spatial information system[J]. Progress in Geography, 2015, 34(2): 129 - 131. (in Chinese)
- [2] 中华人民共和国交通运输部. 2021 年交通运输行业发展统计公报[EB/OL]. (2022 - 05 - 25)[2023 - 03 - 01]. https://xxgk.mot.gov.cn/2020/jigou/zhghs/202205/t20220524_3656659.html.
Ministry of Transport of the People's Republic of China. 2021 transport industry development statistical bulletin[EB/OL]. (2022 - 05 - 25)[2023 - 03 - 01]. https://xxgk.mot.gov.cn/2020/jigou/zhghs/202205/t20220524_3656659.html. (in Chinese)
- [3] GEOFABRIK. OpenStreetMap data extracts [EB/OL]. [2023 - 03 - 01]. <http://download.geofabrik.de/>.
- [4] ALI S M, GUPTA N, NAYAK G K, et al. Big data visualization: tools and challenges [C]//Proceedings of 2nd International Conference on Contemporary Computing and Informatics, 2016.
- [5] KARIM A, SIDDIQA A, SAFDAR Z, et al. Big data management in participatory sensing: issues, trends and future directions [J]. Future Generation Computer Systems, 2020, 107: 942 - 955.
- [6] ÇÖLTEKIN A, GRIFFIN A L, SLINGSBY A, et al. Geospatial information visualization and extended reality displays [M]//Manual of Digital Earth. Singapore: Springer Singapore, 2019: 229 - 277.
- [7] 朱庆, 付萧. 多模态时空大数据可视分析方法综述 [J]. 测绘学报, 2017, 46(10): 1672 - 1677.
ZHU Q, FU X. The review of visual analysis methods of multi-modal spatio-temporal big data [J]. Acta Geodaetica et Cartographica Sinica, 2017, 46(10): 1672 - 1677. (in Chinese)
- [8] GUAN X F, XIE C, HAN L X, et al. MAP-vis: a distributed spatio-temporal big data visualization framework based on a multi-dimensional aggregation pyramid model [J]. Applied Sciences, 2020, 10(2): 598.
- [9] KILSEDAR C E, BROVELLI M A. Multidimensional visualization and processing of big open urban geospatial data on the web [J]. ISPRS International Journal of Geo-Information, 2020, 9(7): 434.
- [10] 杨明奇. 基于 WebGL 的地理信息大数据可视化平台设计与实现 [D]. 北京: 中国科学院大学(中国科学院沈阳计算技术研究所), 2021.
YANG M Q. Design and implementation of geographic information big data visualization platform based on WebGL [D]. Beijing: Institute of Computing Technology, Chinese Academy of Sciences, 2021. (in Chinese)
- [11] RAE A. Geovisualization and geovisual analysis [M]//Handbook of Spatial Analysis in the Social Sciences. Britain: Edward Elgar Publishing, 2022: 322 - 335.
- [12] YANG B, TIAN D, SHAN G H. Tobacco spatial data intelligent visual analysis [J]. Electronics, 2022, 11(7): 995.
- [13] BALLA D, ZICHAR M, TÓTH R, et al. Geovisualization techniques of spatial environmental data using different visualization tools [J]. Applied Sciences, 2020, 10(19): 6701.
- [14] 陆旻, 袁晓如. 地理空间数据可视化中的过滤 [J]. 计算机辅助设计与图形学学报, 2016, 28(5): 702 - 711.
LU M, YUAN X R. Filter in visualization of geospatial data [J]. Journal of Computer-Aided Design & Computer Graphics, 2016, 28(5): 702 - 711. (in Chinese)
- [15] YU J, SARWAT M. Turbocharging geospatial visualization dashboards via a materialized sampling cube approach [C]//Proceedings of IEEE 36th International Conference on Data Engineering, 2020.
- [16] GUO T, FENG K Y, CONG G, et al. Efficient selection of geospatial data on maps for interactive and visualized exploration [C]//Proceedings of the 2018 International Conference on Management of Data, 2018.
- [17] CHEN X, ZHANG J, FU C W, et al. Pyramid-based scatterplots sampling for progressive and streaming data visualization [J]. IEEE Transactions on Visualization and Computer Graphics, 2022, 28(1): 593 - 603.
- [18] TAO W B, HOU X L, SAH A, et al. Kyrix-S: authoring scalable scatterplot visualizations of big data [J]. IEEE Transactions on Visualization and Computer Graphics, 2021,

- 27(2): 401–411.
- [19] LI C H, BACIU G, WANG Y Z, et al. DDLVis: real-time visual query of spatiotemporal data distribution via density dictionary learning [J]. *IEEE Transactions on Visualization and Computer Graphics*, 2021, 28(1): 1062–1072.
- [20] YUAN J, XIANG S X, XIA J Z, et al. Evaluation of sampling methods for scatterplots [J]. *IEEE Transactions on Visualization and Computer Graphics*, 2021, 27(2): 1720–1730.
- [21] HEAVY. AI. Introducing HEAVY. AI 7. 0. [EB/OL]. [2023–03–01]. <https://www.heavy.ai/>.
- [22] ZHOU C, CHEN Z J, PIAN Y Z, et al. A parallel scheme for large-scale polygon rasterization on CUDA-enabled GPUs [J]. *Transactions in GIS*, 2017, 21(3): 608–631.
- [23] ZHANG G M. Detecting and visualizing observation hot-spots in massive volunteer-contributed geographic data across spatial scales using GPU-accelerated Kernel density estimation [J]. *ISPRS International Journal of Geo-Information*, 2022, 11(1): 55.
- [24] NETEK R, MASOPUST J, PAVLICEK F, et al. Performance testing on vector vs. raster map tiles: comparative study on load metrics [J]. *ISPRS International Journal of Geo-Information*, 2020, 9(2): 101.
- [25] WANG G L, CHEN J. A vector data model for efficiently rendering large vector maps on global 3D terrain surfaces [J]. *ISPRS International Journal of Geo-Information*, 2022, 11(4): 234.
- [26] ELDAWY A, MOKBEL M F, ALHARTHI S, et al. SHAHED: a MapReduce-based system for querying and visualizing spatio-temporal satellite data [C]//Proceedings of 2015 IEEE 31st International Conference on Data Engineering, 2015.
- [27] ELDAWY A, MOKBEL M F, JONATHAN C. HadoopViz: a MapReduce framework for extensible visualization of big spatial data [C]//Proceedings of 2016 IEEE 32nd International Conference on Data Engineering, 2016.
- [28] YU J, ZHANG Z S, SARWAT M. GeoSparkViz: a scalable geospatial data visualization framework in the apache spark ecosystem [C]//Proceedings of the 30th International Conference on Scientific and Statistical Database Management, 2018.
- [29] YU J, SARWAT M. GeoSparkViz: a cluster computing system for visualizing massive-scale geospatial data [J]. *The VLDB Journal*, 2021, 30(2): 237–258.
- [30] GHOSH S, ELDAWY A, JAIS S. AID: an adaptive image data index for interactive multilevel visualization [C]//Proceedings of 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019.
- [31] GHOSH S, ELDAWY A. AID*: a spatial index for visual exploration of geo-spatial data [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 34(8): 3569–3582.
- [32] WAN L, HUANG Z, PENG X. An effective NoSQL-based vector map tile management approach [J]. *ISPRS International Journal of Geo-Information*, 2016, 5(11): 215.
- [33] MA M Y, YANG A R, WU Y, et al. DiSA: a display-driven spatial analysis framework for large-scale vector data [C]//Proceedings of the 28th International Conference on Advances in Geographic Information Systems, 2020.
- [34] MA M, WU Y, OUYANG X, et al. HiVision: rapid visualization of large-scale spatial vector data [J]. *Computers & Geosciences*, 2021, 147: 104665.
- [35] LIU Z B, CHEN L, YANG A R, et al. HiIndex: an efficient spatial index for rapid visualization of large-scale geographic vector data [J]. *ISPRS International Journal of Geo-Information*, 2021, 10(10): 647.
- [36] CHEN L, LIU Z B, MA M Y. Interactive visualization of geographic vector big data based on viewport generalization model [J]. *Applied Sciences*, 2022, 12(15): 7710.