

基于定位数据的全景超分辨图像交互可视化框架

王晨泽¹ 沈雪豪^{1,2} 黄振立³ 王政霞¹

¹(海南大学计算机科学与技术学院 海口 570228)

²(海南大学信息与通信工程学院 海口 570228)

³(海南大学生物医学工程学院 海口 570228)

(1045627154@qq.com)

Interactive Visualization Framework for Panoramic Super-Resolution Images Based on Localization Data

Wang Chenze¹, Shen Xuehao^{1,2}, Huang Zhenli³, and Wang Zhengxia¹

¹(School of Computer Science and Technology, Hainan University, Haikou 570228)

²(School of Information and Communication Engineering, Hainan University, Haikou 570228)

³(School of Biomedical Engineering, Hainan University, Haikou 570228)

Abstract Combining super-resolution localization microscopy with panoramic digital pathology provides a powerful tool for biomedical researchers to observe the subcellular structure of the entire sample, but also brings the challenge of visualization of panoramic super-resolution image. However, current super-resolution image visualization methods are unable to process large-scale localization data, provide high-resolution panoramic images, and interactively visualize panoramic images. In response to the above problems, an interactive visualization framework for panoramic super-resolution images based on localization data is proposed, called PNanoViewer, which aims to achieve rapid interactive visualization of large-scale localization data on ordinary computers. This framework constructs the multi-resolution hierarchical structure of localization data based on the random sampling strategy, and visualizes multi-scale panoramic super-resolution image in an interactive way. Meanwhile, the block strategy and multi-thread parallel strategy are used to process large-scale localization data in batches to prevent memory overflow and speed up processing. Experimental results on tens to hundreds of millions of localization datasets show that the proposed PNanoViewer framework can visualize localization data with any size. Compared this framework with three currently popular super-resolution image visualization methods: PALMsiever, ThunderSTORM, and QC-STORM, PNanoViewer has obvious advantages in terms of data volume, resolution, and speed. At the same time, PNanoViewer also provides a useful exploration for the visualization of large-scale localization data.

Key words interactive visualization; large-scale localization data; panoramic super-resolution image; panoramic digital pathology; multi-resolution hierarchical structure

摘要 超分辨定位成像和全景数字病理结合,为研究人员观察整个样本的亚细胞结构提供了有力工具,同时也带来了全景超分辨图像海量数据可视化的挑战。然而,现有的超分辨图像可视化方法无法处理大规模定位数据、不能提供高分辨全景图像和无法交互可视化全景图像。针对以上问题,提出了一个基于定

收稿日期: 2023-08-01; 修回日期: 2023-12-19

基金项目: 国家自然科学基金项目(82160345); 海南省重大科技计划项目(ZDYF2021GXJS017); 海口市重点科技项目(2021-016)

This work was supported by the National Natural Science Foundation of China (82160345), the Key Research and Development Program of Hainan Province (ZDYF2021GXJS017), and the Key Science and Technology Plan Project of Haikou (2021-016).

通信作者: 王政霞 (zxiawang@hainanu.edu.cn)

位数据的全景超分辨图像交互可视化框架,称为 PNanoViewer,旨在普通计算机上实现大规模定位数据的快速交互可视化。该框架基于随机采样策略构建定位数据的多分辨率层级结构,以交互方式可视化多尺度全景超分辨图像;同时采用分块策略和多线程并行策略,分批次处理大规模定位数据,既防止内存溢出又加快了处理速度。从数千万到数亿个定位点数据集上的实验结果表明,PNanoViewer 框架能够可视化任意规模的定位数据。将该框架与目前 3 种流行的超分辨图像可视化方法 PALMsiever, ThunderSTORM, QC-STORM 从数据量、分辨率和速度 3 个方面进行对比,PNanoViewer 都具有明显优势。同时也为大规模定位数据的可视化提供了一个有益的探索。

关键词 交互可视化; 大规模定位数据; 全景超分辨图像; 全景数字病理; 多分辨率层级结构

中图法分类号 TP391

超分辨定位成像(super-resolution localization microscopy, SRLM)克服了光学衍射极限,能在纳米尺度上成像生物结构,使得病理医生和生物学家能够观察细胞的超微结构(如小细胞器、生物大分子等)以及寻找分子标志物的分布特征^[1-2](如细胞分化特征、免疫复合物沉积情况等)。近年来,Creech 等人^[3]开始尝试将 SRLM 用于数字病理研究中,从分子水平揭示病变组织的病理生理学,促生了超分辨数字病理学这个新领域。例如,2021 年 Cordron 等人^[4]利用 SRLM 对 α 突触核蛋白病理成像,揭示了 Lewy 小体的内部组织,为更全面地了解常见的神经系统疾病提供了新机会。全景数字病理^[5]通过对整张病理切片进行扫描成像,可展示病理组织的全貌。全景超分辨数字病理结合大视场和高分辨率的优点,便于探查亚细胞结构整体形态、探测重要的低概率事件和对亚细胞结构进行无偏的研究等。

传统地,病理医生使用显微镜观察和探索切片的多尺度形态特征,从而确认切片中的所有病灶点。具体来说:病理医生首先在低倍镜下查看组织切片,对病变区域进行初步的判断;然后逐渐切换到高倍镜下,进一步识别可疑病变信息;最后,多次重复上述过程,找出切片中所有的病灶点。相应地,全景数字病理通过图像缩放技术(交互可视化)寻找病灶点。即病理医生先在低分辨率图像上搜索病变部位,再在高分辨率图像上查看病变的细节信息。因此,病理图像交互可视化是全景数字病理诊断必不可少的组成部分。全景病理图像具有尺寸大和容量大的特点,由于计算机屏幕和内存的限制,无法一次性加载最高分辨率图像。因此,采用交互可视化策略^[6]每次只按需加载部分图像数据。这不仅可以减少内存加载压力,而且可以解决文件存储和带宽之间的矛盾,实现快速可视化,方便医生浏览整张全景图像^[7]。

目前生成全景超分辨图像有 2 种方式:一种是基

于渲染图像;另一种是基于定位数据。前者是将所有单视场定位数据先以固定分辨率渲染成超分辨图像(样本局部图像),然后使用图像拼接软件如 MIST^[8], BigStitcher^[9], TrakEM2^[10]将其拼接成全景超分辨图像。后者是先使用定位数据拼接方法,如 NanoStitcher^[11], 将所有单视场定位数据拼接成一个全景定位数据表,然后再将其可视化成全景图像。对于前者图像拼接获得的全景超分辨图像,往往使用图像可视化领域常用的图像金字塔模型^[12-13]分层组织并可视化图像,但是由于需要存储多个层级的全景图像,占用存储空间大,造成设备负担重。而定位数据相比于渲染图像能显著降低数据量。例如,要获得视场为 1 mm×1 mm 的全景超分辨(super-resolution, SR)图像,需要 100 个 100 μm ×100 μm 的超分辨图像。当我们以 10 nm 像素尺寸渲染超分辨图像生成 7 层图像金字塔时,最终文件大小为 79 GB,而相应的定位表文件大小为 43 GB。并且随着金字塔层级的增加,图像文件大小会持续增加,而定位表文件大小则保持不变。此外,由于渲染造成的遮蔽效应和信息损失^[14],基于图像的后处理可能会带来不可靠的结果。而直接基于定位表数据的后处理则可利用所有原始信息^[15],能够得到生物学特征更可靠的定量解释。现有的超分辨数据分析方法倾向于使用定位数据获取有关生物结构分布、大小、形状、空间组织和其他类别的定量信息。因此,基于全景定位表数据的可视化可为观察和分析全景超分辨图像带来新机遇。

目前,很多 SRLM 处理软件都能可视化定位表,如 ThunderSTORM^[16], PALMsiever^[17], SMoLR^[18], Live-SIMBA^[19], RainbowSTORM^[20], QuickPALM^[21], QC-STORM^[22] 等。这些软件在数秒到数分钟内就可以完成单视场定位数据(数十万、数百万个定位点)的可视化。但是在面对全景定位数据(数千万、数亿个定位点)时,这些软件就显得力不从心。它们需要数小

时才能渲染出全景定位表，并且可能由于全景定位表的数据量超过内存大小，导致软件崩溃或无法运行，不能获得病理医生诊断需要的全景超分辨图像。特别地，QC-STORM 虽然能够可视化数千万甚至数亿定位点，但是它是以牺牲渲染图像分辨率为代表的，也就是无法提供高分辨率图像。这导致医生可能因无法查看病灶的细节特征，而影响诊断准确度。因此，需要对定位数据的可视化进行深入研究，即在保证图像分辨率的前提下，在普通计算机上可视化大规模定位数据，让研究人员可交互式观察和分析全景超分辨图像。

通过剖析现有的 SRLM 可视化软件的运行机制，我们发现，PALMsiever 等软件(除 QC-STORM)需要一次性将整个定位表加载到内存处理。因此，受限于内存和运行环境的不同，能够处理的最大定位数据量不同，导致无法处理超过计算机内存的定位数据。而 QC-STORM 虽然可以分批加载数据，渲染生成整幅图像，但由于算法自身的限制，如果渲染图像超过 2 GB，软件就会报错，不再继续加载和渲染。同时 QC-STORM 也无法实现交互可视化。因此，针对大规模定位数据，迫切需要探索新的可视化机制，以满足全景超分辨图像可视化及后期分析需求。

受大规模点云可视化策略^[23]的启发，本文提出了一个基于定位数据的全景图像可视化框架，称为 PNanoViewer。该框架首先构建定位数据的多分辨率层级结构，其次采用分块和多线程并行策略，使得用户可以在普通计算机上交互可视化全景超分辨图像，并在大规模定位数据集验证了 PNanoViewer 的有效性。与现有的代表性可视化软件相比，本文方法能在保证图像分辨率的前提下处理任意数据量的定位数据。可视化速度方面，在固态硬盘(solid state drive, SSD)上，PNanoViewer 相比于 PALMsiever, ThunderSTORM, QC-STORM，可视化等待时间分别缩短为原来的 1/57, 1/41 和 1/10。

本文的主要贡献总结有 3 个方面：

1) 基于随机采样策略构建大规模定位数据的多分辨率层级结构，每次可视化按需调度部分定位数据，自适应渲染生成图像，不仅减少内存加载，而且可以解决图像尺寸和计算机屏幕尺寸之间的矛盾，实现快速交互可视化。这解决了用户在普通计算机上无法查看大尺寸全景超分辨图像的问题，使得病理医生可以在普通计算机上探索整个样本的生物结构特征。

2) 采用分块和多线程并行策略，分批次处理定

位数据，具有良好的可扩展性，既防止内存溢出又加快了处理速度。这不仅能够保证大规模定位数据在普通计算机上被流畅地处理，还能够可视化不同数据量大小的定位数据，克服了在普通计算机上无法处理大规模定位数据的难题。

3) 动态交互可视化，可以帮助用户以交互方式快速浏览大规模定位数据的多分辨率层级图像，方便用户观察和分析样本的多尺度生物结构特征。

1 相关工作

本文通过分析大规模定位数据可视化特征，发现大规模定位数据可视化可以借鉴大视场图像可视化和大规模点云可视化 2 个方向的技术：一方面，大规模定位数据通常由多个单视场定位数据组成，通过图像渲染和图像拼接技术得到全景超分辨图像，这样就可以利用已有的大视场图像可视化方法对其进行分析和展示。另一方面，大规模定位数据也可以看作是空间中分布的点集合，类似于扫描仪获取的点云数据。因此，大规模定位数据可视化也可以借鉴大规模点云可视化的思想及技术，如多层次表示、空间划分等。

1.1 大视场图像的可视化

大视场图像可视化是一种使用多分辨率图像层级在计算机屏幕上显示超出屏幕分辨率的图像处理技术，广泛应用于大视场病理图像、天文图像、遥感图像等大规模数据的交互可视化^[24-27]。例如，Stathonikos 等人^[28]开发了大视场数字病理系统，使得医生在普通计算机上即可查看大视场数字病理图像。Pielawski 等人^[29]开发了 TissUUmaps 3，用于在常规计算机上实时查看大规模空间组学数据的多分辨率图像。

在大视场超分辨图像领域，利用多分辨率层级图像技术，可以将高分辨率的细节信息嵌入到低分辨率的全局信息中，从而实现对内存超限的图像进行有效地可视化和处理。总的来说，多分辨率层级技术为大视场超分辨图像在普通计算机上可视化提供了重要的参考依据。

1.2 大规模点云的可视化

大规模点云可视化技术是指利用多分辨率层级结构(level of detail, LOD)技术，将传感器采集的海量点云数据可视化成图像，直观地呈现给用户。LOD 技术可分为 3 类：1) 可编辑 LOD^[30]。它是指层级结构不仅可用于可视化，也可进行增删数据点和点云编辑操作。2) 无冗余 LOD^[31]。它是指在层级结构中不重复

保存数据点, 通过在低层级块上新增点生成高层级块, 其优点是可视化时计算机内存消耗较小. 3)连续 LOD^[32]. 它实现了层级间连续的分辨率变化, 提供更好的视觉效果, 但它对计算机性能有更高的要求.

此外, 随着点云数据量的增大, 利用分布式、GPU 并行、CPU 多线程等技术加快 LOD 构建成为一个研究热点. 其中, Schütz 等人^[23]提出了一种可在普通计算机上快速构建海量点云 LOD 的方法, 该方法比现有方法快 1 个数量级. LOD 策略有助于解决普通计算机内存与定位数据庞大数据量之间的矛盾.

通过上述分析, 大规模数据无论是图像还是点云可视化, 通常采用多分辨率层级思想. 超分辨定位成像的定位数据既可以看作一种特殊形式的点云, 也可以通过渲染生成图像, 因此可以借鉴这 2 种大规模数据的多分辨率层级思想和技术进行可视化处理. 我们对超分辨定位数据和图像进行分析, 发现相比于多分辨率层级图像, 利用定位数据生成多分辨率层级结构, 既能够显著降低存储量, 也易于和基于定位数据的后处理算法相结合, 提高可视化方法的可扩展性.

为了在普通计算机上有效可视化大规模定位数据, 本文借鉴了 Schütz 等人^[23]提出的快速构建 LOD 的思想, 结合定位数据自身的特点, 设计了一种适用于大规模定位数据的层级结构框架, 进而提出了基于定位数据的全景超分辨图像交互可视化方法.

2 数据和方法

2.1 大型全景超分辨图像交互可视化框架

本文提出的全景超分辨图像可视化框架由 3 个部分组成, 分别是多分辨率层级结构构建、可视数据确定以及可视数据渲染. 概括来说, 首先, 对定位数据自下而上随机采样构建多分辨率层级结构, 并存储到磁盘中(图 1 中蓝色虚线框). 然后, 在交互可视化过程中获取每帧图像所需的定位点(可视定位数据), 并准确地从层级结构中调度相应可视定位数据(图 1 中绿色虚线框). 最后, 通过渲染算法将确定的可视定位数据渲染成超分辨图像(图 1 中黄色虚线框).

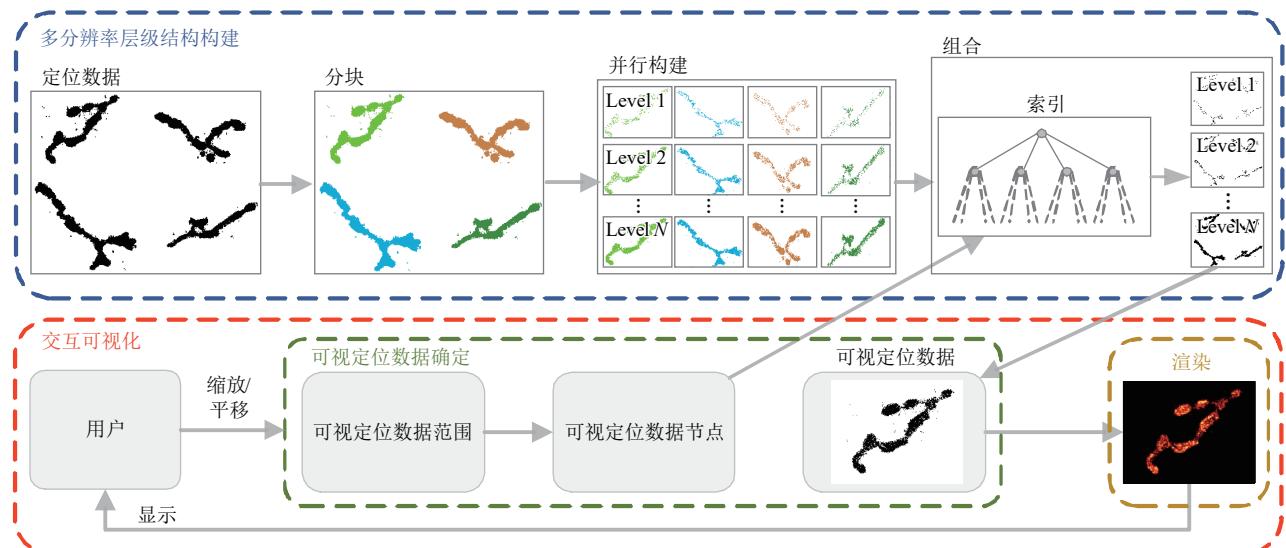


Fig. 1 The interactive visualization framework for large panoramic super-resolution images

图 1 大型全景超分辨率图像交互可视化框架

2.2 多分辨率层级结构的构建

为了在普通 PC 上能够清晰展示全景图像, 本文首先构建定位数据的多分辨率层级结构, 解决大尺寸全景图像由于数据量超出 PC 内存可能无法展示的问题. 构建多分辨率层级结构包括 3 个步骤: 1) 分块. 将大规模定位数据分割成若干个子块. 2) 索引. 并行构建每个块的局部多分辨率层级结构. 3) 合并. 将所有局部多分辨率层级结构合并为一个全局多

分辨率层级结构.

1) 分块. 为了避免定位数据的数据量超出 PC 内存余量导致程序崩溃, 本文采用批处理方式, 将定位点分批次载入到 PC 处理. 即使用分块技术将定位数据按空间和定位点数目划分成多个数据块, 进而控制每次载入到 PC 的定位点的数据量. 图 2 展示了整个分块过程:

步骤 1. 以定位数据的范围为界, 划分出 $2^{level-1} \times 2^{level-1}$

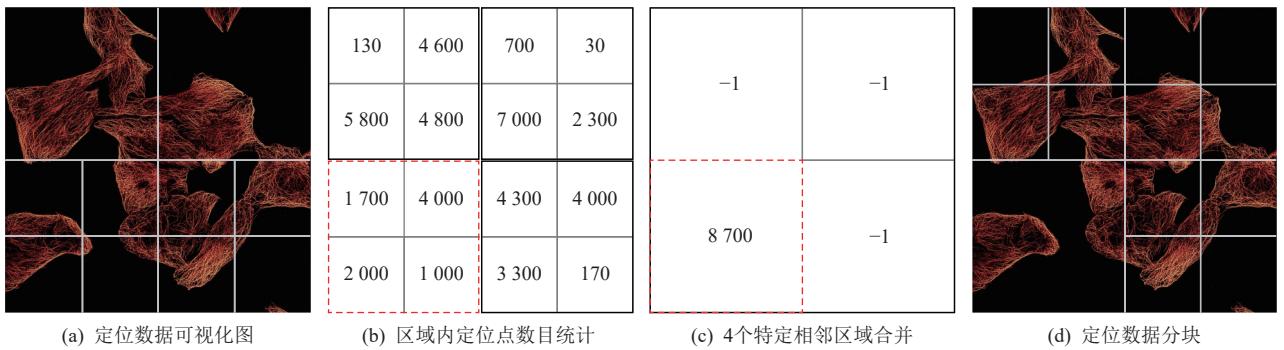


Fig. 2 The process of locating data blocks

图 2 定位数据分块过程

个均等区域,如图 2(a). *level* 为定位数据层级结构的假定初始层级,根据定位点数量经验设置一个初始值,确切的层级在构建完成全局层级结构后确定(定位点数量的经验初值设置,主要参考计算机内存大小和 CPU 性能)设置数据块最大值的原则是在不同性能的计算机上程序都能正常运行.此外,我们鼓励用户根据计算机性能动态设置数据块临界值.

步骤 2. 统计每个区域内包含的定位点数目,如图 2(b).

步骤 3. 根据区域内定位点数目,判断该区域是否能成为数据块.判断规则为: 2×2 相邻区域内(如图 2(b)中的左下虚线框内 4 格)定位点数目之和是否大于阈值(本文设置的定位点数目阈值大小为 1000 万).如果大于阈值,则 2×2 相邻区域都被视为数据块.然后,数据块以其在层级结构中的位置作为文件名存储到磁盘.

步骤 4. 合并 2×2 相邻区域,若区域已被视为数

据块,则标记为“-1”,如图 2(c).否则,将 2×2 相邻区域定位点之和视为合并区域内的定位点数目,如图 2(c)中左下虚线框内格.

步骤 5,重复步骤 3 和步骤 4,直至所有区域都成为数据块,如图 2(d).

注意,在大规模定位数据分割时,需要合理设置数据块大小.一方面,数据块要小到能够多个同时载入到内存中,便于多线程处理.另一方面,数据块不能过小,否则会导致构建层级结构的速度变慢.本文设置阈值为 1000 万,即 2×2 相邻区域内定位点数目之和超过这个值,4 个区域都被视为数据块.除此之外,定位点每个占用 48 B,所以每个 2×2 相邻区域至少占用约 458 MB.本文设置阈值为 500 MB,也就是如果某个区域超过 500 MB,则会继续分割,避免数据块过大.

2)索引.经分块过程生成的单个数据块的数据量远小于 PC 内存,单个数据块的处理只利用了 CPU

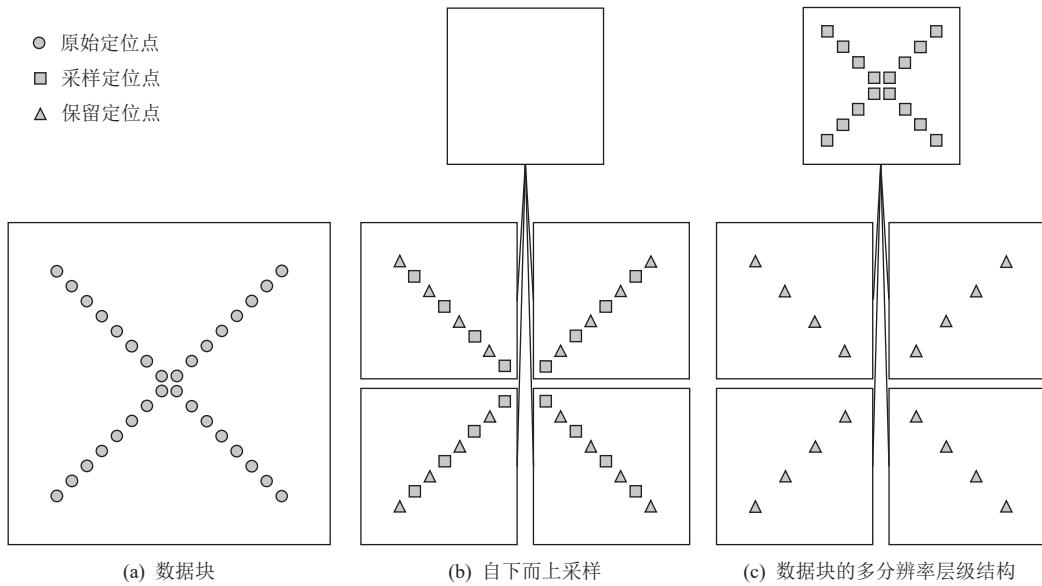


Fig. 3 A single data block builds a multi-resolution hierarchy

图 3 单个数据块构建一个多分辨率层级结构

的部分计算资源。为充分利用内存和CPU资源，我们采用多线程并行技术，同时处理多个数据块。

本文中单个数据块的层级结构使用四叉树组织：首先，将数据块内的定位点分配到四叉树的叶子节点，如图3(b)；然后，从叶子节点开始，自下而上逐层采样填充层级结构中较上层的其他节点，直至数据块层级结构四叉树的根节点，如图3(c)。这里需要注意的一个细节是我们仅将子节点集中被选为采样点的定位点（图3(b)中的方形）存储到父节点，其余定位点（图3(b)中的三角）保留到子节点中，整体节省了存储空间。同时，在可视化子节点区域时，我们调度父节点及祖先节点中在区域内的采样点来保证不会降低子节点渲染图像的分辨率。每个数据块的多分辨率层级结构的构建完成后，数据块中的数据完整地保存在四叉树层级结构中，此时，删除磁盘上相应的数据块。多线程并行构建数据块四叉树层级结构的过程是多个线程中的每个线程同时处理单个数据块。

3) 合并。定位数据层级结构（全局四叉树）是由所有数据块构建的层级结构（局部四叉树）构成。由于局部层级结构的根节点已被标识为其在全局层级结构中的位置，按照“自下而上，从左到右”的合并原则，首先根据标识将局部层级结构链接到全局层级结构；然后，填充全局层级结构根节点与局部层级结构根节点之间的节点，构建全局层级结构；最后，用层级数据文件、索引文件和元数据文件，将全局层级结构的信息存储到磁盘，供交互可视化时调用。层级结构中的定位点数据以节点为单位存储到层级数据文件中。层级结构中节点的索引信息包括节点在层级数据文件中的位置和大小等被存储到索引文件中。层级结构的层级、定位数据的边界范围等信息存储

到元文件中。3个文件的数据存储格式为：层级数据文件和索引文件使用二进制格式存储以减少磁盘空间的占用；元数据文件将层级、定位数据的边界范围等信息和其对应值，以键值对形式存储在json文件中。

2.3 可视数据的确定

在交互可视化过程中，用户通过鼠标移动或缩放操作切换到不同视域。视域改变需要重新渲染相应的可视数据。本节用4个步骤从大规模的层级结构数据中准确地获取可视数据。

步骤1. 计算用户视域对应的多分辨率层级。多分辨率层级的初始值为0，代表最低层。当用户缩小视域时，层级增加1，直至最高层。当用户放大视域时，层级减小1，直至最低层。用户移动视域，层级保持不变。

步骤2. 计算可视数据范围。可视数据范围由其中心位置坐标 $c_{(x,y)}$ 和范围大小确定。 $c_{(x,y)}$ 的初始值为定位数据空间范围的中心位置坐标。当用户进行缩放操作时， $c_{(x,y)}$ 保存不变。当用户进行移动操作时， $c_{(x,y)}$ 跟随用户移动方向和距离以改变其值。可视数据范围大小与层级有关，随层级的增加或减小，以2倍比例缩小或放大。可视数据范围的具体实现公式为

$$\begin{aligned} \min_{(x,y)} &= c_{(x,y)} - border \times 2^{level-1}, \\ \max_{(x,y)} &= c_{(x,y)} + border \times 2^{level-1}, \end{aligned} \quad (1)$$

其中 $\min_{(x,y)}$ 是可视数据范围的最小值， $\max_{(x,y)}$ 是可视数据范围的最大值。 $border \times 2^{level-1}$ 是可视数据范围大小， $border$ 是大规模定位数据范围大小， $level$ 是视域对应的多分辨率层级。

步骤3. 查找与视域范围相交的可视节点。四叉树节点存储该层特定区域内的定位数据，如节点“r00”包含位于四叉树第3层第1棵子树第1个区域内的定位数据，见图4(c)。通过比较视域范围与节点

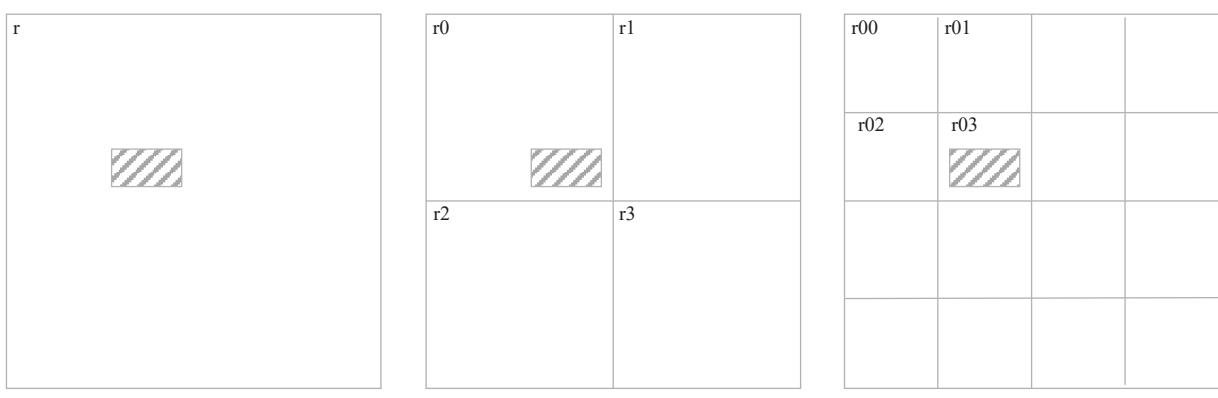


Fig. 4 Relation between the field of view and the visual nodes of each layer

图4 视域与各层可视节点之间的关系

范围，即可找出当前层级与视域范围相交的节点。此外，为节省存储空间，在构建四叉树时，当前层级节点中的部分定位数据已存入到上层祖先节点。因此，祖先节点也是可视节点，查找出的可视节点结果，如图4所示。假定多分辨率层级共有3层，即 $level=2$ ，视域范围为阴影矩形框，从图4(a)~(c)可知，与可视定位数据范围相交的节点分别为“r”“r0”“r03”。

步骤4. 确定可视定位数据。首先，通过利用索引文件中节点的位置信息，构建出组织层级结构的四叉树，并且四叉树的每个节点都包含其定位数据在层级数据文件中的位置和大小信息。然后，根据可视节点名称匹配四叉树中的节点，进而得到可视节点的定位数据在层级数据文件中的位置和大小信息。最后，从层级数据文件中准确地可读取可视节点的定位数据。由于可视节点的范围定位数据大于或等于视域范围，需要进一步根据视域范围从可视节点的定位数据中筛选出可视定位数据。

2.4 可视数据的渲染

获取可视数据后，可以选择任意一种现有的定位点渲染方法将可视数据渲染成超分辨图像，本文选择经典的高斯渲染方法^[1]。

图像是像素值的集合，因此渲染的关键是将定位点信息转换成像素值。高斯渲染首先将每个定位点的位置、定位精度、峰值强度信息转换成一个以定位点位置为中心的 7×7 范围内的强度值数组：

$$I_{(i,j)} = A e^{-\frac{(i-x_0)^2 + (j-y_0)^2}{2\sigma^2}}, \quad (2)$$

其中 $I_{(i,j)}$ 是高斯点扩散函数 (point spread function, PSF) 像素格内的强度值， (x_0, y_0) 为高斯 PSF 中心位置坐标， A 为定位点的峰值信号强度， σ 是定位点的定位精度。

然后，根据定位点在图像上的位置，在图像上累加所有定位点的 7×7 范围内的强度值数组得到一张强度超分辨图像。最后，为了用户更好地观察图像，我们将强度超分辨图像进行伪彩色处理，得到彩色超分辨图像。

2.5 定位数据的描述

本文使用的定位表是通过单分子定位成像技术对U2OS细胞成像和定位获得，每个定位表对应着2维显微镜台的单视场信息，视野约为 $100 \mu\text{m} \times 100 \mu\text{m}$ ，其包含的定位点数量一般是几百万个，每个定位点用12个参数表示。除去2维可视化需要使用的光斑峰值强度(photon)、光斑坐标x(单位pixel)、坐标y(单位pixel)、X方向定位精度CRLBx(单位nm)和Y方向定位精度CRLBy(单位nm)这5个参数外，本文还保留了光斑坐标z(单位nm)、光斑X方向PSF宽

度PSFSigmaX(单位pixel)、光斑Y方向PSF宽度PSFSigmaY(单位pixel)、光斑总强度Total-intensity(单位photon)、光斑背景光子数background(单位photon)、峰值信噪比(signal to noise ratio, SNR)、光斑所在原始图像的帧数frame等参数。这些参数虽然不用于可视化，但可能应用于超分辨图像的后期分析，如聚类、共定位等数据分析。本文使用的定位数据是通过NanoStitcher算法拼接单视场定位表而成。

3 实验与结果

本节定量地比较了所提的PNanoViewer与其他典型定位表可视化软件PALMsiever, ThunderSTORM, QC-STORM的可视化能力。

1) PALMsiever

PALMsiever是一个用Matlab语言开发的定位数据可视化软件。该软件可视化时，需要将全部定位数据一次载入内存，然后生成图像，支持的最大图像分辨率是 4096×4096 pixel。

2) ThunderSTORM

ThunderSTORM是一个功能齐全的超分辨定位成像处理软件。ThunderSTORM在ImageJ上运行，可视化时，也需要将全部定位数据一次载入内存并用窗口显示给用户。

3) QC-STORM

QC-STORM也是一个集成到ImageJ中的超分辨定位成像处理插件。该软件大部分功能使用统一计算设备架构(compute unified device architecture, CUDA)开发，充分利用GPU多线程快速处理的优势。QC-STORM可视化时，可从磁盘中分批次加载定位数据，经CPU传输到GPU中处理，然后生成图像。QC-STORM支持的最大图像容量为2GB。

本文首先利用NanoStitcher拼接了不同视场的定位表数据(从 1×1 个 ROI 到 10×10 个 ROI)，如表1所示，对应的数据量范围是 $0.69 \sim 42.96$ GB。接着从定位表数据量、图像分辨率和等待时间3个方面分析了各对比方法的性能。最后测试了PNanoViewer在交互可视化不同结构(微管、线粒体和膜蛋白)数据集时的表现。

表1列出了本文实验所使用的定位数据视场、定位点数量和数据量大小(二进制形式)。 $M \times N$ 定位数据由 M 行 N 列的相邻 ROI 使用 NanoStitcher 插件拼接获得。整个实验过程在一台工作站完成，配置为：内存32 GB, Intel i7-7800X(6核), 500 GB 三星 970 SSD,

Table 1 Localization Data Information Used in Experiments

表 1 实验中使用的定位数据信息

定位数据	数据量大小/GB	定位点数量	视场/ μm^2
1×1	0.69	15 407 780	100×100
2×2	1.62	36 311 477	182×185
3×3	3.26	73 029 607	242×282
4×4	6.08	136 030 818	370×367
5×5	9.58	214 368 647	459×458
6×6	14.00	314 793 299	542×543
7×7	20.92	468 491 928	608×632
8×8	26.65	596 636 617	702×718
9×9	33.64	752 418 552	808×810
10×10	42.96	960 925 523	897×898

2 TB ST2000DM005-2CW102 硬盘和 11 GB 内存的 NVIDIA GeForce RTX 2080 Ti 显卡。

3.1 定位数据量对比

本节采用最大可处理的定位数据的数据量作为评价指标, 该指标反映了可视化软件或方法能够处理的定位数据的数据量上限。图 5 展示了 PNanoViewer 与其他方法 PALMsiever, ThunderSTORM, QC-STORM 在处理大规模定位数据时的数据量对比结果。

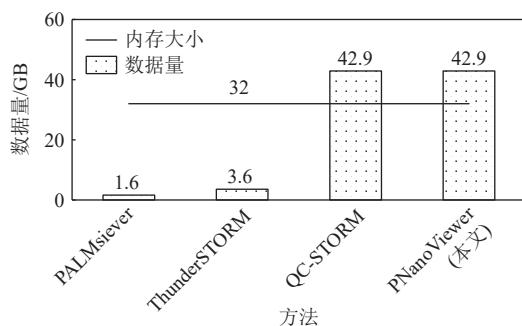


Fig. 5 Maximum data volume of localization data processed by the four methods

图 5 4 种方法处理的定位数据的最大数据量

从图 5 中可以看出, PALMsiever 和 ThunderSTORM 只能处理不超过内存 32 GB 的定位数据, 而 QC-STORM 和 PNanoViewer 则可以处理。这是因为 PALMsiever 和 ThunderSTORM 可视化定位数据时需要一次性将全部数据加载到内存, 而 QC-STORM 和 PNanoViewer 采用了分批次策略处理定位数据。特别是 PNanoViewer 使用空间计数方法, 把大规模定位数据分割成许多远小于内存大小的数据块, 然后, 分批次

处理这些小数据块, 全部数据块处理完毕即完成了大规模定位数据处理。

图 6 展示了各个规模的定位数据与它们分割后的数据块大小之间的关系。各个定位数据的数据块大小平均值保持在 190 MB 附近, 并且最大的数据块一直小于阈值 500 MB, 数据块大小与内存大小相差 1~2 个数量级, 因此这些数据块能够分批载入内存处理。此外, 当计算机的内存容量更小或更大时, 可通过调整阈值控制数据块大小。

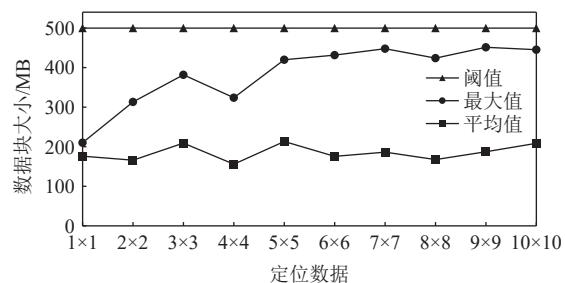


Fig. 6 Size of data blocks after segmentation of localization data with different scales

图 6 各个规模定位数据分割后的数据块大小

3.2 渲染图像分辨率对比

我们评估了 PNanoViewer 和对比方法 PALMsiever, ThunderSTORM, QC-STORM 在表 1 定位表数据集上可提供的图像分辨率。图 7 展示了以 100 nm, 50 nm, 10 nm 像素尺寸渲染表 2 中定位数据得到的图像分辨率, 不同像素大小可以提供观察样本通常需要的粗略轮廓图和清晰细节图。

如图 7 所示, 除了 PNanoViewer, 其余 3 个对比方法均不能查看大规模定位数据的高分辨率图像。其中, PALMsiever 和 ThunderSTORM 因为不能处理大规模定位数据, 所以不能可视化大规模定位数据。而 QC-STORM 虽然能处理大规模定位数据, 但其设置可视化生成的最大图像不超过 2 GB, 即最大图像分辨率不超过 $2 \text{ GB}/(4 \text{ B/pixel}) \approx 537 \text{ MP}$ (megapixel, MP)。大规模定位数据的高分辨率图像远超过 537 MP(例如, 大规模 10×10 定位数据的 10 nm 渲染像素大小的图像分辨率为 8 055 MP), 因此 QC-STORM 不能呈现大规模定位数据的高质量图像, 如图 7(c)。注: 图 7(c)之所以没有 PALMsiever 的实验结果, 是因为 PALMsiever 的最大图像分辨率为 $4 096 \times 4 096 \text{ pixel} \approx 16 \text{ MP}$, 而从 1×1~10×10 定位数据中, 尽管 1×1 的渲染图像分辨率最低, 当以 10 nm 像素大小渲染时得到的图像分辨率约为 120 MP, 远大于 16 MP, 因此 PALMsiever 无法提供 1×1~10×10 任意定位数据的 10 nm 渲

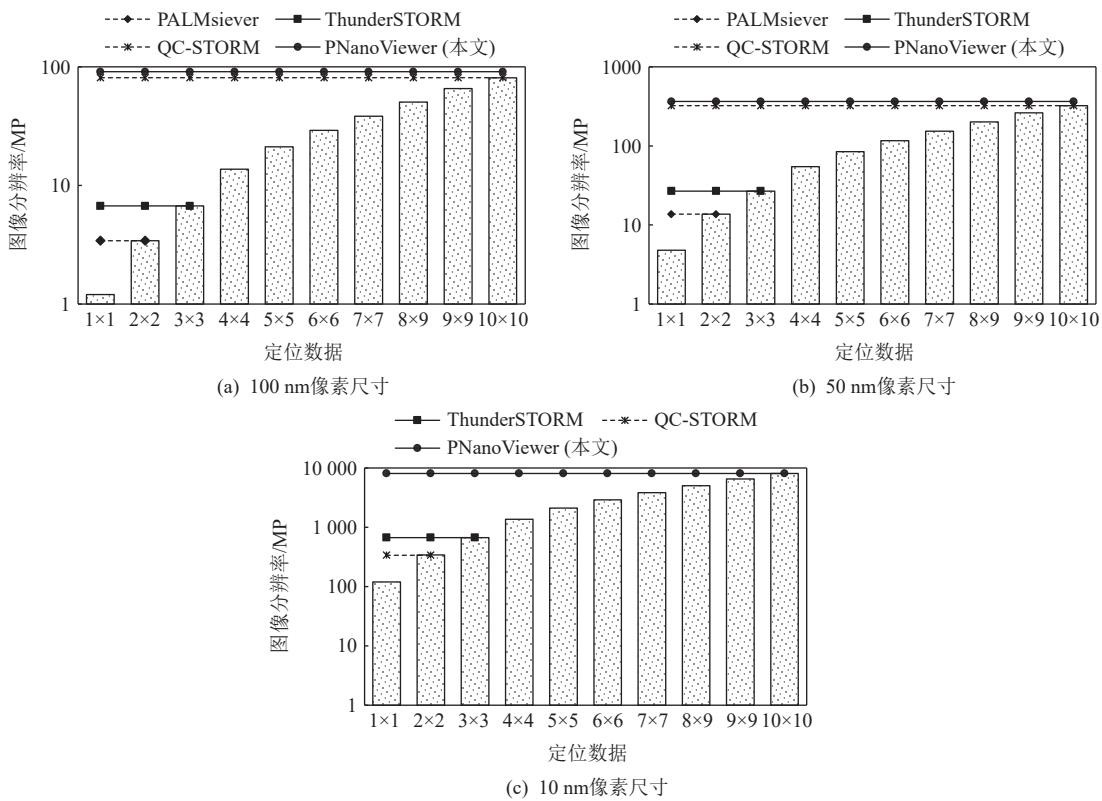


Fig. 7 Image resolution of localization data rendered at different pixel sizes

图 7 定位数据以不同像素尺寸渲染的图像分辨率

Table 2 Visualization Waiting Time for Four Methods

表 2 4 种方法的可视化等待时间

s

渲染像素尺寸/nm	定位数据	固态硬盘				机械硬盘			
		Ps	TS	QS	PV	Ps	TS	QS	PV
100	1x1	404	128	18	7	422	121	16	14
	3x3		393	87	15		411	86	80
	10x10			4 352	256			4 340	3 982
50	1x1	407	135	22	7	425	140	17	14
	3x3		413	141	15		433	138	80
	10x10			10 200	256			9 780	3 982
10	1x1		207	68	7		208	65	14
	3x3		635		15		641		80
	10x10				256				3 982

注: Ps: PALMsiever; TS: ThunderSTORM; QS: QC-STORM; PV: PNanoViewer.

染图像。

相比于 PALMsiever, ThunderSTORM, QC-STORM 方法是将所有定位数据一次性可视化生成整张图像。而 PNanoViewer 是一种基于层级结构的可视化框架, 该框架将定位数据分为不同的层级, 每层包含不同数量和细节的数据点。在实验中, 本节将计算机画布的分辨率设为 1 024×1 024 pixel。用户首先看到的是最低层的全景图像, 它展示了所有定位数据的概览。当用户进行缩放或平移操作时, PNanoViewer 会根据当前画布的范围和分辨率从相应层中提取定位数据,

并生成 1 024×1 024 pixel 的图像进行可视化。通过这种方式, PNanoViewer 可以实现任意渲染像素大小图像的可视化。此外, 用户还可以通过调整画布分辨率来适应不同性能的计算机。因此, PNanoViewer 既能够实现交互式处理又能可视化大规模定位数据。

3.3 等待时间对比

3.1~3.2 节的实验说明本文所提 PNanoViewer 比 PALMsiever, ThunderSTORM, QC-STORM 方法具有更好的可视化能力。本节继续测试 PNanoViewer 在不同渲染像素尺寸下(100 nm, 50 nm, 10 nm)的等待时间。

可视化等待时间是指用户从发起可视化请求到看到可视化结果所需的时间。表2展示了4种方法的可视化等待时间，结果展示PNanoViewer的可视化等待时间是最短的，除QC-STORM在机械硬盘(hard disk drive, HDD)上以100 nm渲染 10×10 数据集外。其中PNanoViewer分别比PALMsiever, ThunderSTORM, QC-STORM快了($404\sim 7$ s)/ 7 s ≈ 57 倍、($635\sim 15$ s)/ 15 s ≈ 41 倍、($68\sim 7$ s)/ 7 s ≈ 9 倍。PNanoViewer能够快速可视化的主要原因是它采用了多线程技术来提高可视化效率。多线程技术可以充分利用计算机的多核处理器资源，将数据处理分配给不同的线程并行执行，从而减少了单个线程的负担，大幅度减少处理时间。此外，在表2中，PNanoViewer在SSD上速度最大提升了($3982\sim 256$ s)/ 256 s ≈ 15 倍，而其他方法速度变化不大，最大仅0.05倍，这表明PNanoViewer更能充分利用SSD高性能读写数据的特点。PNanoViewer有更频繁的内存与磁盘间的数据传输，SSD的高速读写会缩短数据传输时间，相比于HDD，PNanoViewer可视化时间有明显提升。

PNanoViewer的另一优势是可视化同一定位表的不同分辨率图像花费的等待时间相同。这是因为它利用了多分辨率层级技术来提高可视化效率。由于构建多分辨率层级所花费的时间远远大于交互可视化调度和渲染定位数据所花费的时间。因此，构建完成定

位数据的多分辨率层级结构后，每次可视化只需要从层级结构中调度部分定位数据，就可以快速查看多分辨率图像。而其他方法每次查看不同分辨率图像都需要重新加载和可视化定位数据，需要消耗大量时间。

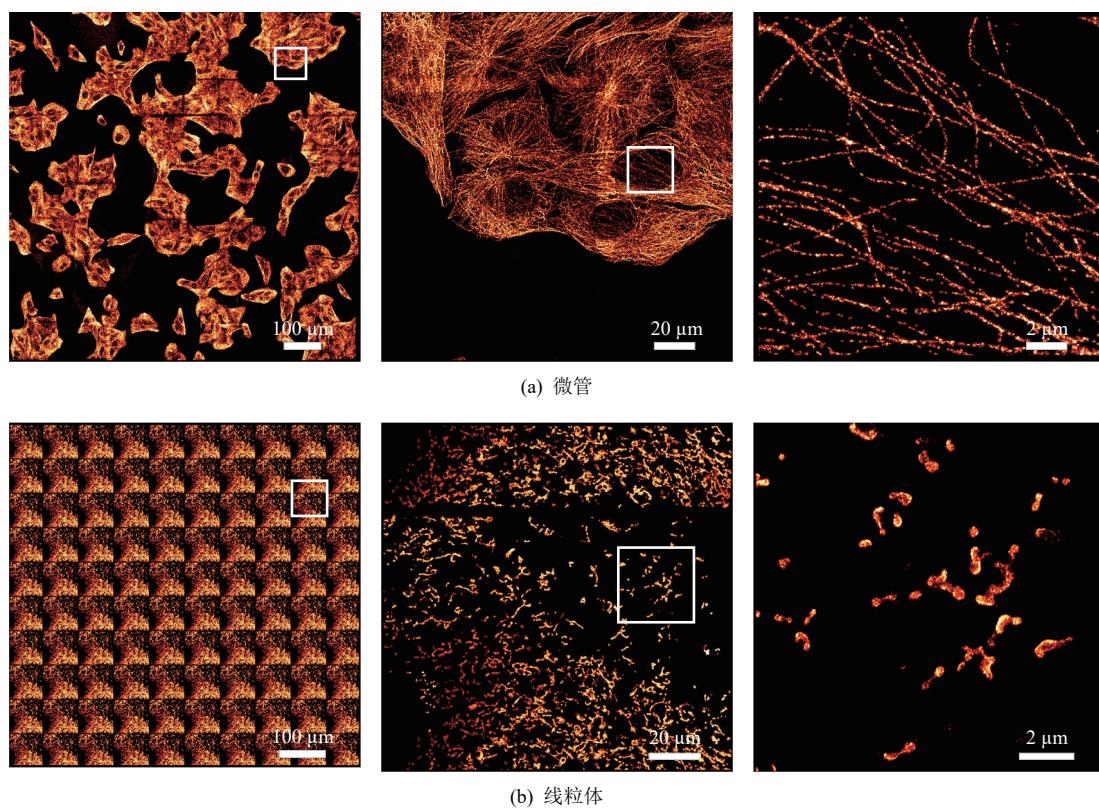
3.4 不同生物结构的可视化

为了展示PNanoViewer可视化框架在多种生物结构(微管、线粒体和膜蛋白)的大规模定位数据上的表现，本节使用了3个大规模数据集：微管定位数据(见表1中 10×10)、线粒体定位数据以及膜蛋白定位数据(见表3)。表3中，线粒体和膜蛋白定位数据是通过将原始数据在横向和纵向上复制平移而得到的。

Table 3 Information on Mitochondrial and Membrane Protein Localization Data Based on 10×10 Scale
表3 线粒体和膜蛋白 10×10 视场的定位数据信息

生物结构	数据量大小/GB	定位点数量	视场/ μm^2
线粒体	17.7	40 378 840	720×700
膜蛋白	12.2	27 870 730	510×500

图8中每个分图的第1列展示了微管、线粒体和膜蛋白在PNanoViewer最低分辨率层级可视化生成的全景图，用户可通过查看全景图观察到生物结构的样貌、分布及形态等特征。在此基础上，用户进



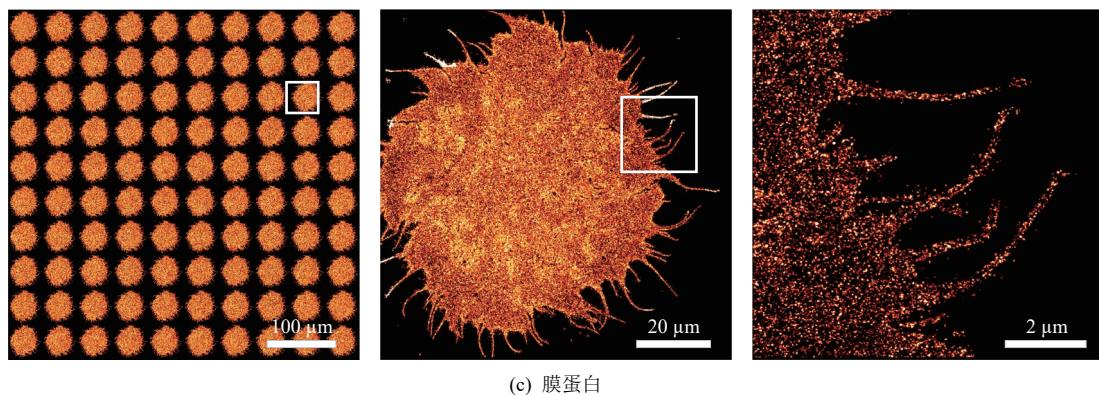


Fig. 8 Visualization of results at multiple resolution levels for large scale microtubule, mitochondrial and membrane protein localization data
图 8 大规模微管、线粒体和膜蛋白定位数据的多分辨率可视化结果

行放大操作后, 视域缩小, 即观察到的生物结构区域减少, 细节增多如第 2, 3 列. 反之, 用户进行缩小操作后, 视域增大, 能看到更广阔区域的生物结构概貌. 另外, 用户可进行移动操作, 在同一分辨率下观察不同区域生物结构的特征. 总的来说, 无论在哪种生物结构的大规模定位数据上, PNanoViewer 都能够实现高效的交互可视化, 探索生物结构的多尺度特征.

4 总 结

本文发展了一个大型全景超分辨图像交互可视化框架 PNanoViewer. 该框架解决了大规模定位数据由于超出 PC 内存无法被普通 PC 直接处理或无法高分辨率显示的问题. 首先, 构建了大规模定位数据的多分辨率层级结构, 并且以交互方式调用层级结构中的可视数据, 从而大幅度减少每次可视化需要渲染的数据量, 快速生成超分辨图像. 其次, 在构建层级结构过程中采用分块和多线程并行策略同时处理多个定位数据块, 实现了大规模定位数据的快速可视化. 使用不同规模的实验数据, 本文比较了 PNanoViewer 与 PALMsiever, ThunderSTORM, QC-STORM 这 3 种经典的超分辨图像可视化方法的性能. 我们发现 PNanoViewer 能处理大数据量定位表(即使定位表数据量超出 PC 内存大小)并且能够快速可视化, 尤其在 SSD 上, PNanoViewer 相比于 PALMsiever, ThunderSTORM, QC-STORM, 可视化等待时间分别缩短为原来的 1/57, 1/41 和 1/10. 更重要的是, PNanoViewer 能够可视化大型全景高分辨图像(10×10 ROI 以 10nm 像素尺寸渲染), 而 PALMsiever, ThunderSTORM, QC-STORM 无法实现. 同时我们也在不同生物结构上(微管、线粒体、膜蛋白)的图像上验证了 PNanoViewer

可视化的鲁棒性. 当然, 我们提出的 PNanoViewer 仍有一些局限性, 其中最主要的一个局限是低分辨率全景超分辨图像会有信息损失, 未来我们计划开发一种更有效的层级结构构建策略, 利用所有的定位点数据信息生成低分辨率图像, 保证信息不丢失的同时不影响可视化速度.

作者贡献声明: 王晨泽提出研究思路, 完成实验并撰写论文; 沈雪豪参与论文修改与结果分析; 黄振立指导论文框架设计并审查论文; 王政霞指导论文框架和实验方案设计, 并修改论文.

参 考 文 献

- [1] Lelek M, Gyparaki M T, Beliu G, et al. Single-molecule localization microscopy[J]. *Nature Reviews Methods Primers*, 2021, 1(1): 1–39
- [2] Fuhrmann M, Gockel N, Arizono M, et al. Super-resolution microscopy opens new doors to life at the nanoscale[J]. *Journal of Neuroscience*, 2022, 42(45): 8488–8497
- [3] Creech M K, Wang Jing, Nan Xiaolin, et al. Superresolution imaging of clinical formalin fixed paraffin embedded breast cancer with single molecule localization microscopy[J]. *Scientific Reports*, 2017, 7: 40766
- [4] Codron P, Letournel F, Marty S, et al. Stochastic optical reconstruction microscopy (STORM) reveals the nanoscale organization of pathological aggregates in human brain[J]. *Neuropathology Applied Neurobiology*, 2021, 47(1): 127–142
- [5] Hanna M G, Ardon O, Reuter V E, et al. Integrating digital pathology into clinical practice[J]. *Modern Pathology*, 2022, 35(2): 152–164
- [6] Balzer C, Oktavian R, Zandi M, et al. Wiz: A web-based tool for interactive visualization of big data[J]. *Patterns*, 2020, 1(8): 100107
- [7] Zhang Xinhua, Li Caiwei, Zhang Yu, et al. Multi-target panoramic digital pathology: From principle to application[J]. *Chinese Optics*, 2022, 15(6): 1258–1274 (in Chinese)

- (张新华, 李才伟, 张瑜, 等. 多靶点全景数字病理: 从原理到应用[J]. *中国光学*, 2022, 15(6): 1258–1274)
- [8] Chalfoun J, Majurski M, Blattner T, et al. MIST: Accurate and scalable microscopy image stitching tool with stage modeling and error minimization[J]. *Scientific Reports*, 2017, 7(1): 4988–4998
- [9] Hoerl D, Rusak F R, Preusser F, et al. BigStitcher: Reconstructing high-resolution image datasets of cleared and expanded samples[J]. *Nature Methods*, 2019, 16(9): 870–874
- [10] Cardona A, Saalfeld S, Schindelin J, et al. TrakEM2 software for neural circuit reconstruction[J]. *Plos One*, 2012, 7(6): 1–8
- [11] Du Yue, Wang Chenze, Zhang Chen, et al. Computational framework for generating large panoramic super-resolution images from localization microscopy[J]. *Biomedical Optics Express*, 2021, 12(8): 4759–4778
- [12] Hausen R, Robertson B E. FitsMap: A simple, lightweight tool for displaying interactive astronomical image and catalog data[J]. *Astronomy and Computing*, 2022, 39: 100586
- [13] Cao Shixiang, Jiang Jie, Zhang Guangjun, et al. Multi-resolution image stitching of edge feature points[J]. *Journal of Computer Research and Development*, 2011, 48(9): 1788–1793 (in Chinese)
(曹世翔, 江洁, 张广军, 等. 边缘特征点的多分辨率图像拼接[J]. 计算机研究与发展, 2011, 48(9): 1788–1793)
- [14] Wu Yule, Tschanz A, Krupnik L, et al. Quantitative data analysis in single-molecule localization microscopy[J]. *Trends in Cell Biology*, 2020, 30(11): 837–851
- [15] Khater I M, Nabi I R, Hamarneh G. A review of super-resolution single-molecule localization microscopy cluster analysis and quantification methods[J]. *Patterns*, 2020, 1(3): 100038
- [16] Hagen G M, Kiek P, Ovesn M, et al. ThunderSTORM: A comprehensive ImageJ plugin for PALM and STORM data analysis and super-resolution imaging[J]. *Bioinformatics*, 2015, 30(16): 2389–2390
- [17] Pengo T, Holden S J, Manley S. PALMsiever: A tool to turn raw data into results for single-molecule localization microscopy[J]. *Bioinformatics*, 2015, 31(5): 797–798
- [18] Paul M W, de Gruiter H M, Lin Zhanmin, et al. SMoLR: Visualization and analysis of single-molecule localization microscopy data in R[J]. *BMC Bioinformatics*, 2019, 20(1): 1–7
- [19] Li Hongjia, Xu Fan, Gao Shan, et al. Live-SIMBA: An ImageJ plug-in for the universal and accelerated single molecule-guided Bayesian localization super resolution microscopy (SIMBA) method[J]. *Biomedical Optics Express*, 2020, 11(10): 5842–5859
- [20] Davis J L, Soetikno B, Song Kihee , et al. RainbowSTORM: An open-source ImageJ plug-in for spectroscopic single-molecule localization microscopy (sSMLM) data analysis and image reconstruction[J]. *Bioinformatics*, 2020, 36(19): 4972–4974
- [21] Henriques R, Lelek M, Fornasiero E F, et al. QuickPALM: 3D real-time photoactivation nanoscopy image processing in imageJ[J]. *Nature methods*, 2010, 7(5): 339–340
- [22] Li Luchang, Xin Bo, Kuang Weibing , et al. Divide and conquer: Real-time maximum likelihood fitting of multiple emitters for super-resolution localization microscopy[J]. *Optics Express*, 2019, 27(15): 21029–21049
- [23] Schütz M, Ohrhallinger S, Wimmer M. Fast out-of-core octree generation for massive point clouds[J]. *Computer Graphics Forum*, 2020, 39(7): 155–167
- [24] Schüffler P J, Stamelos E, Ahmed I, et al. Efficient visualization of whole slide images in web-based viewers for digital pathology[J]. *Archives of Pathology Laboratory Medicine*, 2022, 146(10): 1273–1280
- [25] Lajara N, Luis Espinosa-Aranda J, Deniz O, et al. Optimum web viewer application for DICOM whole slide image visualization in anatomical pathology[J]. *Computer Methods and Programs in Biomedicine*, 2019, 179: 104983
- [26] Wang Hongkun, Cao Yi, Xiao Li. Image-based interactive visualization of large-scale data sets[J]. *Journal of Computer Research and Development*, 2017, 54(4): 855–860 (in Chinese)
(王弘堃, 曹轶, 肖丽. 基于图像的大规模数据集交互可视化[J]. 计算机研究与发展, 2017, 54(4): 855–860)
- [27] Yu Haijun, Li Shengyang, Zhang Tao. A space-time interactive visualization approach for managing remote sensing data[J]. *Journal of Information Science Engineering*, 2021, 37(2): 395–412
- [28] Stathonikos N, Nguyen T Q, Spoto C P, et al. Being fully digital: Perspective of a dutch academic pathology laboratory[J]. *Histopathology*, 2019, 75(5): 621–635
- [29] Pielawski N, Andersson A, Avenel C, et al. TissUUmaps 3: Interactive visualization and quality assessment of large-scale spatial omics data[J]. *Biorxiv*, 2022, 9(5): 1–14
- [30] Park S, Ju S, Yoon S, et al. An efficient data structure approach for bim-to-point-cloud change detection using modifiable nested octree[J]. *Automation in Construction*, 2021, 132: 103922
- [31] Deibe D, Amor M, Doallo R. Supporting multi-resolution out-of-core rendering of massive LiDAR point clouds through non-redundant data structures[J]. *International Journal of Geographical Information Science*, 2018, 33(3): 1–25
- [32] van Oosterom P, van Oosterom S, Liu Haicheng, et al. Organizing and visualizing point clouds with continuous levels of detail[J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2022, 194: 119–131



Wang Chenze, born in 1996. Master candidate. His main research interest includes image processing.

王晨泽, 1996年生. 硕士研究生. 主要研究方向为图像处理.



Shen Xuehao, born in 1995. PhD candidate. His main research interests include deep learning and image processing.

沈雪豪, 1995年生. 博士研究生. 主要研究方向为深度学习、图像处理.



Huang Zhenli, born in 1976. PhD, professor, PhD supervisor. His main research interests include digital diagnosis and optical microscopy technique.
黄振立, 1976生.博士, 教授, 博士生导师. 主要研究方向为数字诊疗、光学显微成像技术.



Wang Zhengxia, born in 1977. PhD, professor, PhD supervisor. Her main research interests include medical image processing and super resolution optical image processing.
王政霞, 1977生.博士, 教授, 博士生导师. 主要研究方向为医学图像处理、超分辨光学图像处理.