项目结题验收单

专家验收表（主持人所在单位组织 3-5 名专家对项目进行验收、自评。）

<table>
<tr><td>项目名称</td><td colspan="5">图书馆座位智能管理系统的设计与实现</td></tr>
<tr><td>主持人</td><td colspan="2">席亚军</td><td>职务/职称</td><td colspan="2">副研究馆员</td></tr>
<tr><td>所在单位</td><td colspan="5">塔里木大学图书馆</td></tr>
<tr><td>专
家
意
见</td><td colspan="5">本项目自立项以来，项目组成员按照各自分工与项目计划安排，顺利完成了本课题项目的研究和实践工作，主要完成了以下工作：
（1）系统的设计与实现。包括管理端和客户端两大模块，管理端负责后台管理，客户端则为用户提供登录、座位预约、通知查看、违约通知查看及个人信息管理等功能。
（2）数据库的设计。使用 MyBaits 通过数据库的账号密码信息来从数据库中获取信息,使用前后端分离技术来进行系统的开发,前端又包含管理端和服务端,管理端有首页,用户信息,图书信息,预约选座信息,公告信息等模块,服务端有登录注册,预约选座,座位查看,个人信息等模块。后端使用 SSM 框架,使用 sql 语句从数据库中获取到信息,在前端展示。
（3）撰写论文 2 篇，其中一篇文 SCI（中科院二区，IF=4.7）；出版专著 1 部；开发系统一个，并获得软件著作权 1 项。

（如需要可增加页数）</td></tr>
<tr><td>专家签字</td><td>何艳香</td><td>杨正华</td><td>刘元梅</td><td>许如兰</td><td></td></tr>
<tr><td>职务/职称</td><td>研究馆员</td><td>副研究馆员</td><td>副研究馆员</td><td>副研究馆员</td><td></td></tr>
</table>

项目编号：2024051

CALIS 全国农学文献信息中心研究项目

结题报告

项目名称：图书馆座位智能管理系统的设计与实现

项目关键词：座位智能管理系统；微信小程序；Vue；SSM

项目单位（盖章）：塔里木大学图书馆

通信地址：新疆阿拉尔市塔里木大学图书馆 843300

项目主持人：席亚军

联系电话：13565142267

电子邮件：187170403@qq.com

提交日期：2025 年 5 月 10 日

图书馆座位智能管理系统的设计与实现

关键词：座位智能管理系统；图书馆座位预约；微信小程序；Vue；SSM；JAVA

1. 研究背景、目的及意义

对于座位资源管理系统的使用者—学生，一方面解决了线上预约和实际座椅使用情况不同步的问题[1]。满足了学生实时掌握各图书馆内座位资源动态情况，根据自身需求喜好对座位的进行个性化选择的需求。另一方面，将图书馆座位资源的管理者转换为学生，即从源头将占座行为的产生者赋予座位资源管理者角色，让其承担座位资源管理者责任，减少了占座行为的产生[2]。对于座位资源管理系统的拥有者—高校图书馆，一方面可以优化座位资源配置，提高座位资源的使用率，高效公正地分配来馆学生。另一方面是在现有设施的基础上进行改进，不仅成本低，还可以发挥资源的最大利用率。不仅给学生创造一个更加舒适安静的外部学习环境，缓解占座问题引起的学生与学生之间，学生与图书馆之间的矛盾。还在完善座位管理系统的同时丰富来高校智慧图书馆建设[3]。运用小程序作为载体设计的软硬件结合产品，是对图书馆座位管理系统设计的一个探索，通过创作、探索和总结这一类作品的设计形式、设计方法和创作规律，指导自身的设计实践，发展目前对图书馆座位管理系统的创作研究，也对类似主题作品的创作和研究起到参考和借鉴的作用[4]。

基于上述问题，图书馆座位智能管理系统是一款面向全国高校学生的一站式座位预约和管理的功能性产品，旨在辅助高校图书馆优化座位资源配置。在深入了解高校学生群体需求特征下，结合线上线下的现代服务模式，对高校图书馆中存在的座位资源使用效率低下、学生占座现象频发等问题提出解决方案。首先，它是由微信小程序客户端以及网页管理端组成。在图书馆座位智能管理系统的设计中，要确保系统的功能得到充分的考虑，并且精心设计出一个完善的界面，以便用户可以轻松访问、预约座位。然后，为了提高系统的可操作性，还需要深入研究座位信息和预约信息，并且利用所掌握的技术来实现功能的开发。最后，该系统以 SSM 为主体，以 Idea、VScode、微信开发者工具作为开发工具，使用 MySQL 数据库，JAVA

语言，前端界面使用 Vue 框架，主要实现了用户管理、图书管理、座位管理、预约选座管理、通知公告、意见反馈、修改密码等功能模块。该系统具有简单的接口、容易的操作和容易的维护。

2. 研究内容及方法

2.1 研究方案

本文在大量阅读国内外相关文献和学习资料的基础上[5]，确定研究方案，研究图书馆座位智能管理系统的设计与实现，技术路线如图 1 所示。

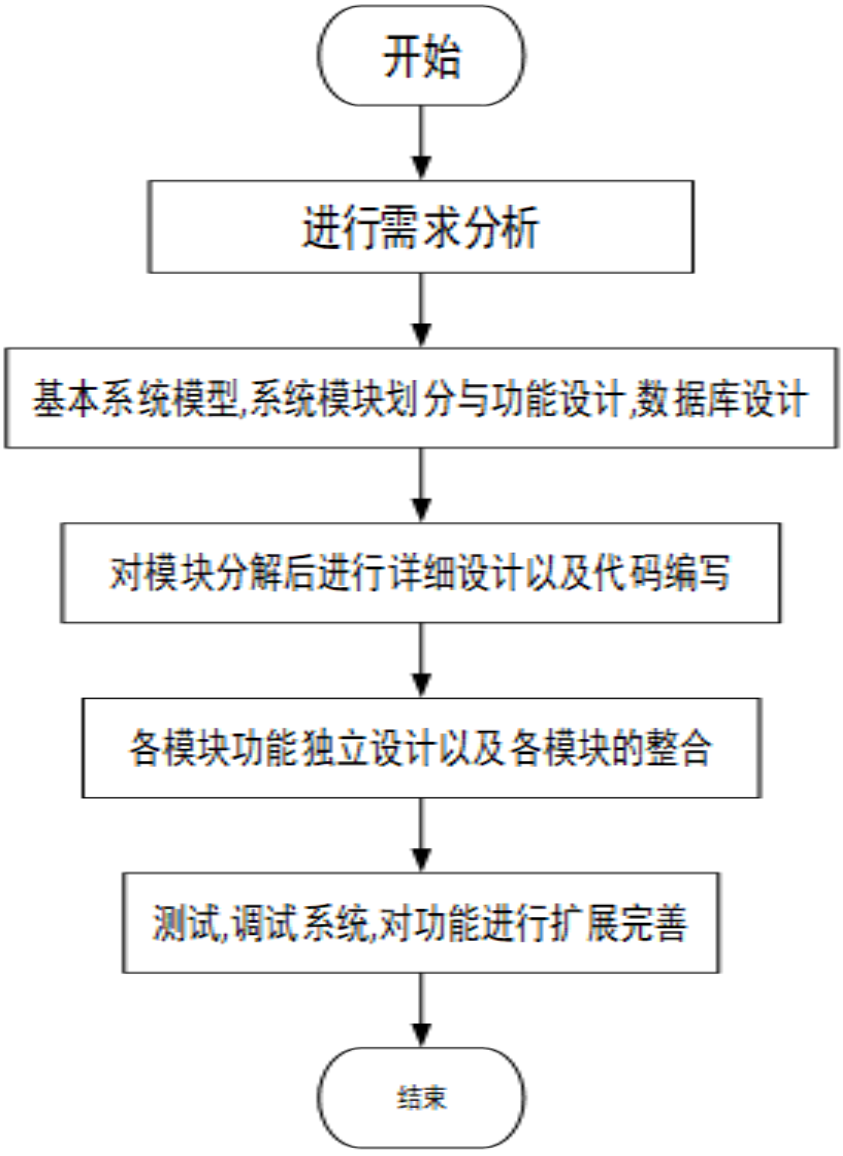


图 1 系统开发流程图

2.2 研究方法

2.2.1 MySQL 数据库

MySQL 被认为是一种高效的数据存储系统，这主要归功于它的开放源代码，这降低了使用的成本，同时也保证了存储的速度。MySQL 还提供多种 API，能够实现跨平台的访问，并且在全球范围内都非常受欢迎[6]。 MySQL 可以被描述为一个具有高度可扩展性、高速运行、支持多种线程和多种使用者的大型数据库系统。

2.2.2 JAVA 语言

Java 还在模块化编程和人工智能领域取得了显著进展[7]。Java 9 引入了模块化系统，更好地支持了组件化开发和代码的重用性。在人工智能领域，Java 提供了丰富的工具和库，如机器学习库 Weka、DL4J 和 Java-XML，为开发者们构建智能化的应用程序提供了便利和支持。对于本系统而言,java 语言完全满足系统的开发需要,所以本系统使用 java 座位后端开发语言。

2.2.3 SSM 框架

SSM 框架，作为一种经过验证的稳定项目框架，其显著优势在于为工程师们提供了一个类似于即插即用的成品结构[8]。因此，众多工程师倾向于选择 SSM 框架来进行工程设计，这不仅极大地节省了他们在创建新应用程序时的时间和精力，还显著提高了开发的便捷性和效率。SSM 框架的随时可用性，以及其为代码重用创造的优良环境，进一步巩固了其在工程界中的首选地位。SSM 是一种由三个开源框架组成的系统，分别是 Spring、 SpringMVC 和 MyBatis。

2.2.4 Vue 框架

Vue 框架是我在前端界面设计中的首选，它的 API 结构清晰易懂，而且操作起来也比较容易，因此 Vue.js 被认为是一款轻量级的 JavaScript 框架[9]。这使得开发者可以快速掌握并投入到图书馆座位智能管理系统的开发中，提高了开发效率。并且 Vue.js 通过提供组件化的开发框架，使得开发者能够将界面分解为更小、更易于操作的单元。在图书馆座位预约系统中，可以将座位展示、预约操作、用户信息等功能模块拆分为不同的组件，使得代码结构更加清晰，便于维护和扩展。采用数字驱动的技术，将数据与视觉形象紧密结合，从而达到双重目的。即，随着数字的改变，视觉形象将及时进行调整；同样，随着视觉形象的改变，数字也将随之改变。

此外，该技术还能够让开发人员集中精力完成任务，从而避免对视觉形象的过度依赖。Vue.js 拥有庞大的生态系统和丰富的插件库，包括 Vue Router（用于构建单页面应用）、Vuex（用于状态管理）、Element UI（前端组件库）等。这些工具和库可以帮助我快速构建功能完善、性能优异的图书馆座位智能管理系统[10-15]。

2.3 研究内容

2.3.1 登录功能

当管理员打开浏览器的首页时，您就会看到一条提示您输入您的账号或密码的登陆界面。这里的所有信息都必须经由正规的审核程序，方可使您的账号被允许使用。在用户提交登录信息后，系统会对用户名和密码进行验证。首先，系统会在数据库中查找输入的用户名是否存在。只有当用户的密码和账号完全匹配，他们才能够成功地登录系统，并享受到相应的权限和访问特权。通过登录功能，不仅可以确保系统的安全性，还可以让用户轻松访问网站，从而更加便捷地完成后续任务。

图 2-1　管理端登录界面

2.3.2 系统首页功能

通过登录后会进入到系统首页的界面，这是用户与系统交互的主要入口。系统首页包含了丰富的功能模块和快捷操作，为用户提供直观、便捷的导航和使用体验。在首页上，用户可以快速查看最新的图书信息，座位信息，预约总数信息，图书类别统计信息，座位分布统计信息以及预约日期统计信息以便及时了解系统的最新动

态。此外，首页还提供了常用的功能按钮和菜单，方便用户快速访问和操作其他功能模块。

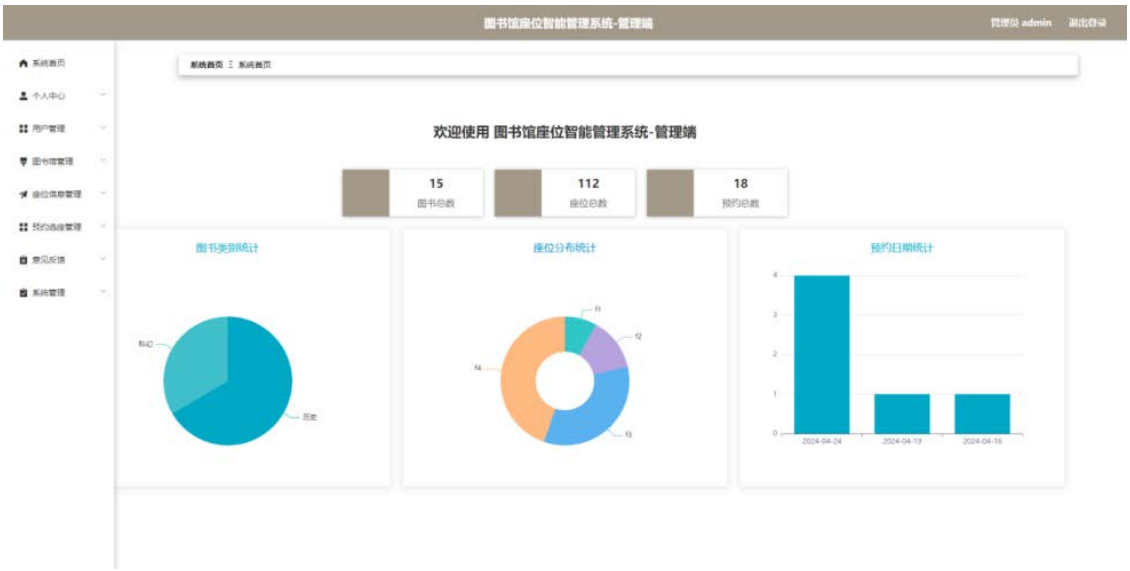


图 2-2　系统首页界面

2.3.3 个人中心功能

在该界面中，管理员能够便捷地修改自己的账号信息以及密码，确保系统安全与个人隐私的双重保障。当管理员意识到密码可能存在泄露风险时，他们也可以在这个界面上迅速地进行密码修改，确保账号安全不受威胁。

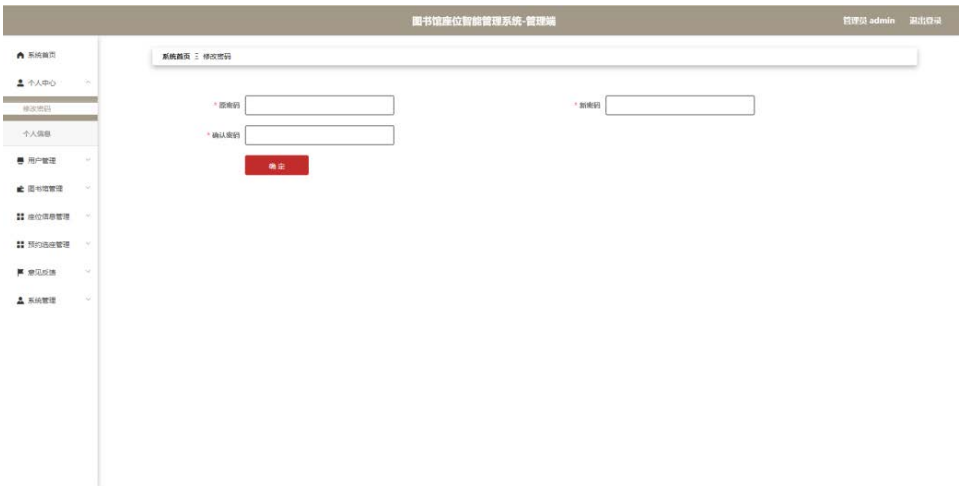


图 2-3　个人中心界面

2.3.4 用户管理功能

点击用户管理栏，管理员可以进入一个全新的界面，这里有许多与用户有关的功能，可以让他们更加轻松地掌握和管理自己的账户。这个界面提供了用户列表的展示，包括用户的基本信息，方便管理员快速了解系统内的用户情况。管理员可以对用户进行新增、编辑、删除等操作，以满足不同用户管理需求。此外，用户管理界面还提供了搜索和筛选功能，方便管理员快速定位并管理特定的用户。

图 2-4　用户信息界面

2.3.5 图书管理功能

与用户管理相同，管理员可以进入该界面来查看书籍列表，获取书本的基本信息，同时可以进行增加，修改，删除书本的操作。此外，该界面提供了搜索框，管理员可以根据需求快速定位想要编辑的图书。

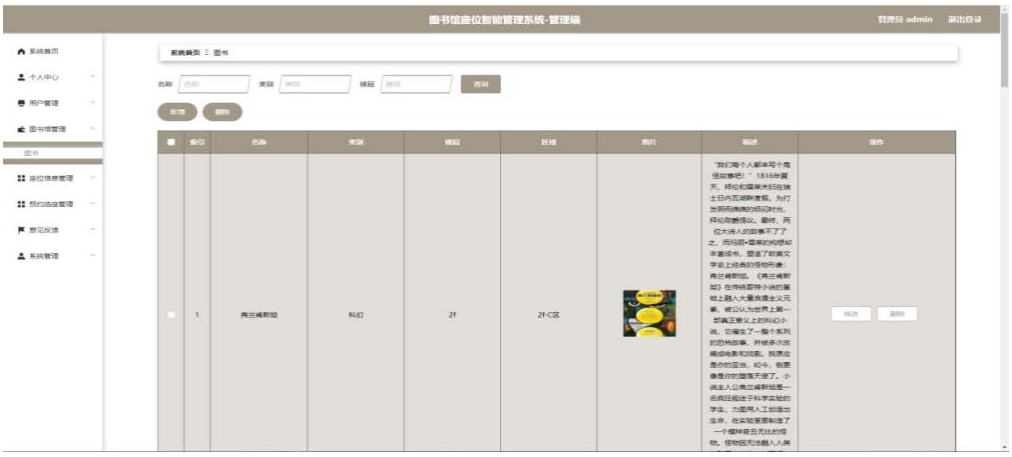


图 2-5　图书信息界面

2.3.6 座位管理功能

与用户管理相同，在这个界面中，管理员可以进行座位的添加、删除和编辑操作，以适应图书馆或会议室等场所的座位变动需求。通过座位管理功能，管理员能够轻松掌控座位资源，提升座位的利用率，为读者或参会人员提供更好的使用体验。

图 2-6　座位信息界面

2.3.7 预约管理功能

通过这个界面，管理员可以更加精确地掌握预约信息，从每一位用户的预约记录中，包括座位、时间、状态等，为用户提供更加完善的服务。此外，管理员还可以根据需求对预约信息进行筛选和搜索，快速定位到特定的预约记录。

图 2-7　预约信息界面

2.3.8 登录注册功能

用户可以通过该功能进入服务端微信小程序，开始他们的在线体验。它确保了用户身份的安全验证和信息的准确性。用户在登录时，需要输入有效的用户名和密码，系统将对其进行验证，确保只有合法用户才能访问小程序内的各项功能[30]。

A 用户登录界面

B 用户注册界面

图 2-8 登录注册界面

2.3.9 小程序首页

当用户登录小程序后，他们就会进入到首页，这里有许多功能可供用户使用。例如，用户可以通过柱状图查看当前时间的座位信息，并且图书推荐栏会实时向用户推荐当前最受欢迎的三本图书，以提供用户最佳的使用体验。

图 2-9 小程序首页界面

2.3.10 座位预约功能

通过座位预约功能，用户可以轻松实现对图书馆座位的在线预约。该功能为用户提供了一个便捷的平台，使他们能够在任何时间、任何地点进行座位预约，避免

了现场排队等待的麻烦。用户只需在系统中选择所需的座位、预约时间和时长，系统将自动为其保留所选座位。同时，用户还可以随时查看座位的预约情况，了解座位的实时状态，以便做出更合理的预约选择。座位预约功能的实现，不仅提高了座位的使用效率，也提升了用户的体验满意度。

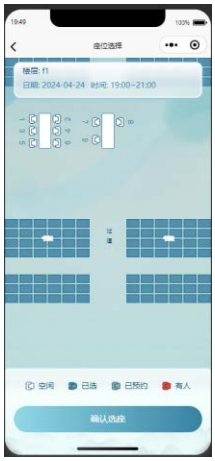


图 2-10　座位预约界面

2.3.11 图书功能

通过该功能用户可以查看书本的详细情况，包括位置，描述等信息。用户只需简单操作，便可获取所需信息，极大地提升了借阅体验，使得图书的查找与选择变得更为便捷与高效。同时，这也为用户提供了更个性化的阅读推荐，助力他们在书海中快速找到适合自己的读物。

图 2-11　图书查看界面

2.3.12 我的座位功能

通过该功能，用户可以方便的查看自己在图书馆的预约座位信息。无论是签到，还是续约，都能一键完成。这一功能不仅提供了座位预约的便捷性，还确保了座位资源的合理分配与高效利用。用户可以根据自己的需求，提前预约座位，避免到图书馆后找不到座位的尴尬情况。

图 2-12　我的座位界面

3. 结论与建议

3.1 结论

经过数月的辛勤努力，本项目已经顺利完成。尽管目前系统功能仅能满足基本需求，设计上也还存在一些不尽如人意之处，但这一过程对本项目来说是极其宝贵的。通过实际操作，本项目深入学习了系统设计与开发中不可或缺的技术知识，这些实践经验对本项目今后的职业生涯具有深远的影响。同时,本项目也逐渐认识到自身在专业技能和问题解决能力上的不足。面对系统设计中的诸多挑战和难题，本项目通过查阅专业网站、参与论坛讨论等方式，积极寻找解决方案，这不仅锻炼了本项目的独立学习能力，也提升了本项目对问题的敏感度和解决能力。当然，本项目的顺利完成离不开指导老师的悉心指导和众多开发者的无私帮助。他们的专业知识和丰富经验为我提供了宝贵的支持，使我能够克服一个又一个难关，最终实现了项目的既定目标。

3.2 建议

回顾项目的全部历程，本人付出了艰辛的努力和汗水，但也从中获得了许多珍贵而难以忘怀的成果。对未来怎样完善系统有了几点展望：

（1）增强系统交互性，便于用户们的操作，本人计划引入一系列新的界面设计和功能。首先，本人将优化用户界面，使其更加直观易用，通过减少操作步骤和增加视觉提示，使用户能够更快地掌握系统。此外，本人还想开发个性化推荐系统，根据用户的使用习惯和偏好，提供定制化的座位选则，以提升用户体验。

（2）优化系统性能，提高用户的并发访问量，在上百个用户同时使用系统时不会导致系统服务崩溃的情况。

在这几个月的开发与探索下，尽管系统的实施存在一些问题，但通过对其进行系统编程的学习，我们拥有了更强的信心，并且相信未来的发展会更加顺利。

4 项目成果

4.1 专著

[1] 席亚军. 现代图书馆信息化建设理论与研究[M]. 西北工业出版社, 2024,5.

4.2 论文

[1] 潘俊璋,席亚军*. 图书馆座位智能管理系统的设计与实现[J]. 科技创新发展, 2024, 1(4): 38-40.

[2] Liu et al. A cotton organ segmentation method with phenotypic measurements from a point cloud using a transformer[J]. Plant Methods, 2025: 21:37.（中科院二区， IF=4.2）

4.3. 软件著作权

[1] 席亚军. 图书馆座位智能管理 APP V1.0. 国家知识产权局, 2024.

5 参考文献

[1] 贾佳.基于物联网的智慧图书馆研究[J].侨园,2020 年 5 期,88.

[2] 蒋谢芳,马璇,王长浩,高健.智慧图书馆座位管理系统设计与实现[J]数字技术与应用,2019 年 6 期,158-159.

[3] 黄元培.网络数据库安全隐患及防范措施解析[J].无线互联科技,2020 年 9 期 28-29.

[4] Yongbin Zhang; Ronghua Liang; Yuansheng Qi; Hejie Chen; Bingran Li.Yongbin Zhang; Ronghua Liang; Yuansheng Qi; Hejie Chen; Bingran Li[J]2021 7th Annual International Conference on Network and Information Systems for Computers (ICNISC).23-25 July 2021.

[5] 高许雷.基于物联网的公共场所座位智能管理系统设计[J].电子测试,2020 年 12 期,77-78.

[6] 刘震.基于 J2EE 的图书馆管理系统设计与实现[M],厦门大学,2013.

[7] 杨钧越.基于 RFID 技术的图书馆座位管理平台设计与实现[M].哈尔滨工业大学,2014.

[8] 宋娜娜.设计模式在管理信息系统开发中的应用研究[J].河南工程学院学报（自然科学）,2010 年 3 期,34-37.

[9] 毛炎.基于 Vue.js 框架的 Web 方言地图的设计与开发[M].武汉大学,2018.

[10] 王海月.基于人工智能的高校图书馆智能化服务设计[J].农业图书情报,2019 年 3 期 75-82.

[11] 谢明诠,林奕纯,曾晟.高校图书馆借助科技手段培养读者行为习惯的探索[J].图书馆建设,2014 年 7 期,36-39.

[12] 刘健冬.面向城乡一体的地籍管理数据库设计研究[J].科技创新导报,2014 年 18 期,177-178.

[13] 束进,牛渝,周巍伟.企业信息化建设中的主数据管理[J].上海船舶运输科学研究所学报.2016 年 1 期,81-84.

[14] 杨胜文,史美林,姜进磊,许骏.面向服务的学习评价网格安全体系[J].清华大学学报（自然科学版）.2005 年 10 期,1425-1428.

[15] 刘瑶.高校图书馆智慧转型中的管理问题研究[M].山东大学,2019.

附录：

1. 专著：席亚军. 现代图书馆信息化建设理论与研究[M]. 西北工业出版社, 2024,5.

http://opac.nlc.cn/F/?request=%E7%8E%B0%E4%BB%A3%E5%9B%BE%E4%B9%A6%E9%A6%86%E4%BF%A1%E6%81%AF%E5%8C%96%E5%BB%BA%E8%AE%BE%E7%90%86%E8%AE%BA%E4%B8%8E%E7%A0%94%E7%A9%B6&func=find-m&find_code=WRD&FIND_BASE=NLC01&FIND_BASE=NLC09

返回结果列表 | 保存/邮寄 | 重新查询

第 1 条记录(共 2 条) < 上一条记录　下一条记录 >

标准格式　文献索取

头标区	-----nam0 22----- 450
ID 号	013514652
通用数据	20250324d2024 em y0chiy50 ea
题名与责任	现代图书馆信息化建设理论与研究 [专著] / 席亚军著
出版项	西安 : 西北工业大学出版社, 2024
载体形态项	124页 ; 26cm
语言	chi
内容提要	本书旨在探索和研究图书馆信息化的理论和实践问题，实现图书馆信息化建设和图书馆服务的现代化。全书主要内容包括图书馆信息化建设的支撑平台与相关技术、大数据环境下图书馆信息化建设与服务转型、数字图书馆建设中的版权保护、智慧图书馆的构成与发展策略以及图书馆智慧服务与智慧图书馆情境感知服务模式。
主题	图书馆工作 -- 信息化建设 -- 研究
中图分类号	G250.7
著者	席亚军 著
所有单册	查看所有馆藏单册信息
馆藏	中文基藏
馆藏	书刊保存本库
馆藏	北区中文图书区


现代图书馆信息化
建设理论与研究
席亚军◎著
西北工业大学出版社

目 录

前 言

随着科学技术的不断发展，信息化已经深刻地影响着我们的生活，也对图书馆建设提出了更高的要求。如何实现图书馆的信息化建设，提高读者借阅体验和管理效率，得到了越来越多图书馆工作者的关注和思考。

图书管理信息化是科学技术发展的必然要求，是满足时代需求的一种自然方式，主要是指信息储存数字化、传输手段网络化、管理控制智能化、人员素质信息化等。即是说借助现代新技术，提高图书管理的自动化水平，使图书管理走向现代化。

本书旨在探索和研究图书馆信息化的理论和实践问题，实现图书馆信息化建设和服务的现代化。全书主要内容包括：现代图书馆信息资源管理、图书馆信息化建设的技术与人才建设、大数据环境下图书馆信息化建设与服务转型、数字图书馆建设中的版权保护、智慧图书馆的构成与发展策略、智慧服务与智慧图书馆情境感知服务模式等。图书馆信息化建设和管理研究是一项系统性、综合性、专业性、前沿性和实用性强的研究，本书语言叙述通俗易懂，应用性强，侧重实用的同时把握理论深度，取舍合理，体系完整，内容先进。

本书是基于知识图谱的图书馆借阅问答系统设计与实现（CALIS 全国农学文献信息中心研究项目，编号：2022057）、图书馆智能服务系统的设计与实现（新疆高等学校图书情报工作委员会科研项目，编号：TGW-20222406）、图书馆座位智能管理系统的设计与实现（CALIS 全国农学文献信息中心研究项目，编号：2024051）研究成果。

本书在撰写过程中得到了许多同行专家的支持和帮助，在此表示衷心的感谢；同时，还参考了很多的相关著作和文献资料，在此向有关作者表示由衷的感谢。

由于作者水平有限，书中难免存在错误或不妥之处，恳请读者批评指正。

著 者

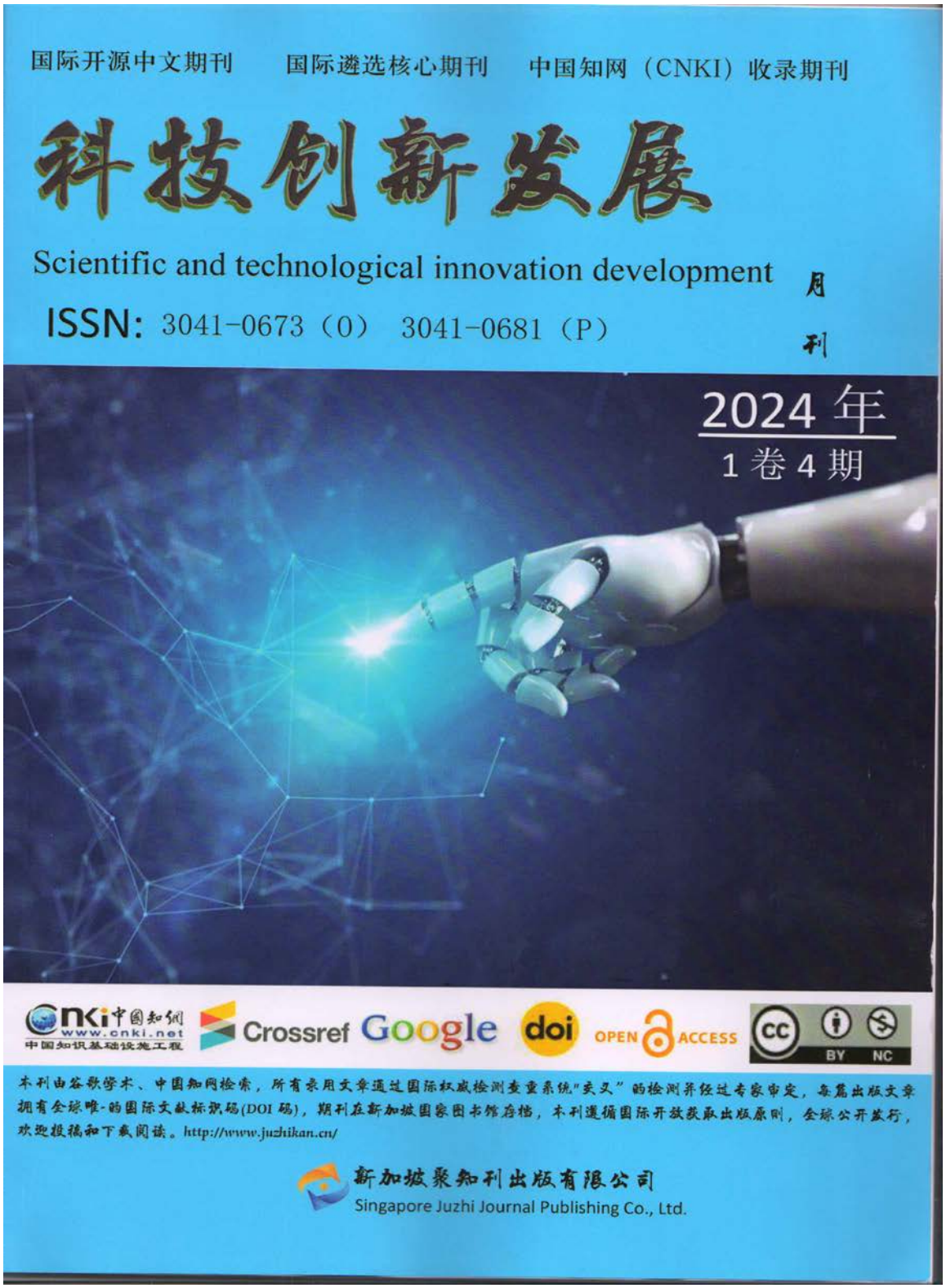
2023 年 11 月

现代图书馆信息化建设理论与研究

ISBN 978-7-5612-9290-7
9 787561 292907
定价：76.00元

2. 论文：

潘俊璋,席亚军*. 图书馆座位智能管理系统的设计与实现[J]. 科技创新发展，2024, 1(4): 38-40.



科技创新发展

Scientific and technological innovation development

2024年第1卷 第4期

主办单位：新加坡聚知刊出版有限公司

国际刊号：3041-0673（O）

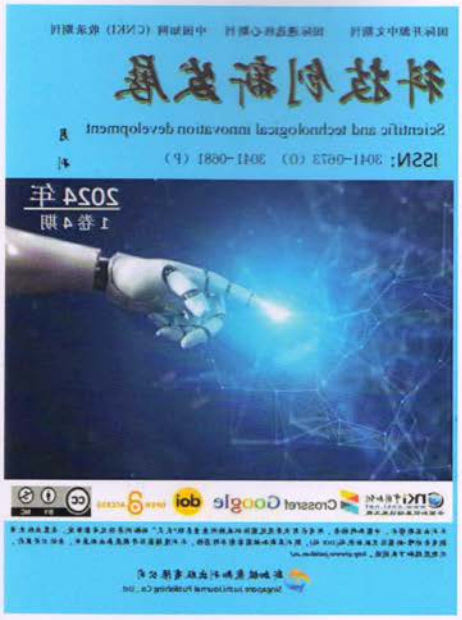
3041-0681（P）

发行周期：月刊

官　　网：http://www.juzhikan.cn/

地　　址：152 BEACH ROAD#11-05 GATEWAY EAST SINGAPORE（189721）

本刊声明



《科技创新发展》本刊长期以来注重质量，编排规范，选稿较严格，学术水平较高，深受高校教师及科研院所研究人员青睐。本刊为开源（Open Access）期刊，出刊的所有文章均可在全球范围内免费下载，中国主流数据收录。

本刊关注科技创新知识共享的国际开源中文期刊，刊登的内容主要围绕科技研究、智慧科技、人工智能、区块链技术、智能制造、金融科技、科技人才培养、行业交流与展示，为推动国内外工程技术和科学技术发展服务。

期刊范围：（人工智能、区块链技术、大数据分析、云计算服务、物联网应用、产业经济、自动化生产、科技教育、智能医疗、绿色科技、创新驱动、数字化转型、科研投入、科技人才培养、科技政策扶持等。）

本刊是一本由同行评审的高要求、高水准的学术期刊出版物，编者鼓励与本刊相关的、有理论和实践贡献的来稿。

目 录

SINGAPORE JUZHI JOURNAL PUBLISHING PTE. LTD

公司地址

152 BEACH ROAD#11-05GATEWAY EASTSINGAPORE

(189721)

http://www.juzhikan.cn.

科技创新发展

扫码查文件

ISSN:3041-0681

04

9 773041 068001

图书馆座位智能管理系统的设计与实现

潘俊璋[1,2] 席亚军[3*]

1. 塔里木大学信息工程学院，新疆阿拉尔，843300；
2. 塔里木绿洲农业教育部重点实验室，新疆阿拉尔了，843300；
3. 塔里木大学图书馆，新疆阿拉尔，843300；

摘要：随着中国的快速经济增长，消费者对移动设备的需求不断增加，许多移动 APP 和软件正在得到广泛的推崇。其中，微信已成为消费者的首选，因此，本文推出了一款新的图书馆座位智能管理系统。首先，它是由微信小程序客户端以及网页管理端组成。在图书馆座位智能管理系统的设计中，要确保系统的功能得到充分的考虑，并且精心设计出一个完善的界面，以便用户可以轻松访问、预约座位。然后，为了提高系统的可操作性，还需要深入研究座位信息和预约信息，并且利用所掌握的技术来实现功能的开发。最后，该系统以 SSM 为主体，以 Idea、VScode、微信开发者工具作为开发工具，使用 MySQL 数据库，JAVA 语言，前端界面使用 Vue 框架，主要实现了用户管理、图书管理、座位管理、预约选座管理、通知公告、意见反馈、修改密码等功能模块。该系统具有简单的接口、容易的操作和容易的维护。

关键词：座位智能管理系统；图书馆座位预约；Vue；SSM
DOI:10.69979/3041-0673.24.4.013

引言

随着中国改革开放以来网络基础设施的迅猛发展和个人电子设备的普及，互联网用户数量激增，人们对计算机网络的依赖程度日益加深，已经逐渐开始尝试使用互联网的方便、快捷、高效的特点去解决生活中的问题[1]。现代管理系统的广泛应用已经取代了许多传统的手工操作，显示出将线下管理转移到线上的巨大优势。因此，图书馆座位智能管理系统的出现不仅反映了时代的需求，也预示着未来发展趋势[2]。图书馆座位预约系统的普及，不仅极大地方便了用户的使用，也标志着图书馆服务现代化的重要步骤。

图书馆座位智能管理系统的实现可以提升图书馆资源利用率：图书馆座位是宝贵的资源，尤其在考试或学习高峰期，座位供不应求。通过预约系统，可以实现对座位资源的合理分配和有效利用，减少座位空置和占用情况，从而最大化满足读者的学习需求。还可以优化用户体验：传统的图书馆座位分配方式往往导致读者在图书馆内寻找座位，浪费时间和精力。图书馆座位智能管理系统可以让读者提前在线预约座位，避免现场寻找座位的麻烦，提高读者的学习效率和满意度。能提升管理员管理效率：图书馆管理人员可以通过预约系统实时监控座位的预约情况，便于进行座位管理和调整。同时，系统还可以记录读者的预约行为，为图书馆服务改进提供数据支持。

由此可见，图书馆座位预约系统对于提升图书馆资源利用率、优化用户体验、提升管理效率、推动图书馆数字化转型以及培养读者自律意识等方面都具有重要意义[3]。因此，选题具有较高的实际应用价值和社会效益。图书馆座位预约系统的研究内容主要围绕实现座位资源的合理分配和高效利用，以满足学生、教师和图书馆管理人员的不同需求。本人在参考了很多高校以及社会中的图书馆的座位管理系统，决定进行图书馆座位智能管理系统的开发和设计工作，旨在帮助图书馆可以更加简洁明了的管理座位，降低管理员的工作量，也可以使用户在使用上获得更好的使用体验。

1 图书馆座位智能管理系统分析

1.1 可行性分析

1.1.1 技术可行性

图书馆座位智能管理系统的开发采用的是 Java 语言，在构建系统时，前端设计与 Vue 框架紧密结合，利用 Vue 的响应式特性和组件化优势来提升用户体验。后端方面，依托了开源的 MySQL 数据库来存储和管理系统数据，其稳定性和广泛的社区支持被广大的开发者们所认同[4]。

1.1.2 技术可行性

因为本次开发主要由不需要任何的开发费用，使用的软件也全都是开源软件，不需要额外的资金购买，在支持方面主要通过询问老师以及网络中的开发者，因此，它在经济上没有太多的投资，也没有太多的负担，所以

它在经济上也是可以实现的。

1.2 需求分析

1.2.1 功能性需求

用户登录与注册功能是每一个系统所必须的功能，用户需通过注册账号并登录系统，以便进行座位预约、查看等操作。而在图书馆座位智能管理系统中，座位查看与选择时最为主要的一个功能，用户应能查看图书馆各区域的座位占用情况，并选择空余座位进行预约。当用户预约完成后，就要进行座位的预约与签到：用户可根据自己的需求预约座位，并在适时进行签到，避免浪费座位资源。系统应当能实时获取座位的占用状态更新，确保用户查询到的信息准确可靠。数据统计与分析，系统应能收集座位使用数据，如预约率、使用率等，并生成报表供管理员查看，系统还应当拥有个人信息的查看与修改功能，便于用户更新自己的个人信息。

1.2.2 功能性需求

系统应具备良好的稳定性，确保在高峰时段也能正常运行，不会报错，满足用户需求。系统的 UI 界面也应当设计的简洁明了，让用户上手即可用，系统还应该拥有较强的安全性，当用户在系统中进行操作时，不会泄露用户的个人信息，系统也应当适当留下扩展接口，让系统可以快速扩展功能。安全也是一个重要的指标，系统要保证用户的信息安全，不会泄露出去。

2 图书馆座位智能管理系统的总体设计

2.1 系统开发流程设计

图书馆座位智能管理系统的开发是一个精心规划的过程，其中管理模块的设计、系统使用的数据库分析、代码编写开发是构建信息管理应用程序的核心三部曲。

2.2 功能模块介绍

用户在该系统中可以使用预约作为，图书查看，我的位置，我的预约，个人信息，意见反馈等多种功能。在该功能图中，用户可以通过查看图书模块查询图书信息、寻找图书、归还图书等；在预约座位模块中，可以查看座位信息、预约座位等，在通知公告模块中，在意见反馈模块中，可以评论提交自己的意见；在我的预约模块中，可以查看已经预约的座位信息；在个人信息模块中，可以修改个人账号的登录密码。通过这些功能，用户可以方便地在该系统中进行图书查看、座位预约、信息查看、交流互动等操作，提高了用户的图书馆使用体验。

2.3 数据库设计

在开发项目中，数据库的设计是非常重要的环节。数据库的合理设计可以直接影响项目的开发和运行效果[5]。一个优秀的数据库设计能够确保数据的准确性、一致性和完整性，提高数据的访问效率，降低系统的维护成本。因此，在开发过程中，我们必须对数据库设计给予足够的重视，充分理解业务需求，明确数据之间的关系，合理设计数据表结构，以及考虑数据的安全性和扩展性[6-7]。

2.3.1 座位信息表

座位信息表用于存储图书馆的座位信息，包括楼层、座位号、创建时间等信息组成，管理员可以通过楼层分布以及座位号来精确查看每一个座位的信息，同时，管理员每新增一个座位就会在小程序服务端的预约选座界面中实时更新。主键 id 用于分辨每一个不同的座位，不至于造成混淆的情况。

2.3.2 用户信息表

用户信息表用于存储用户信息，它由创建时间、账号、密码、姓名、性别、邮箱等信息组成，并通过主键 id 来区分每一个用户，当用户在个人中心中修改信息时，也会调用后端的修改接口在数据库中实时更新。

3 图书馆座位智能管理系统的功能实现

3.1 系统首页功能

通过登录后会进入到系统首页的界面，这是用户与系统交互的主要入口。系统首页包含了丰富的功能模块和快捷操作，为用户提供直观、便捷的导航和使用体验。在首页上，用户可以快速查看最新的图书信息，座位信息，预约总数信息，图书类别统计信息，座位分布统计信息以及预约日期统计信息以便及时了解系统的最新动态。此外，首页还提供了常用的功能按钮和菜单，方便用户快速访问和操作其他功能模块。

3.2 用户管理功能

点击管理者栏，管理员可以进入一个全新的界面，这里有许多与用户有关的功能，可以让他们更加轻松地掌握和管理自己的账户。这个界面提供了用户列表的展示，包括用户的基本信息，方便管理员快速了解系统内的用户情况。管理员可以对用户进行新增、编辑、删除等操作，以满足不同用户管理需求。此外，用户管理界面还提供了搜索和筛选功能，方便管理员快速定位并管理特定的用户。

3.3 座位管理功能

与用户管理相同，在这个界面中，管理员可以进行座位的添加、删除和编辑操作，以适应图书馆或会议室等场所的座位变动需求。通过座位管理功能，管理员能

够轻松掌控座位资源，提升座位的利用率，为读者或参会人员提供更好的使用体验。

4 图书馆座位智能管理系统的小程序服务端功能实现

4.1 登录注册功能

用户可以通过该功能进入服务端微信小程序，开始他们的在线体验。它确保了用户身份的安全验证和信息的准确性。用户在登录时，需要输入有效的用户名和密码，系统将对其进行验证，确保只有合法用户才能访问小程序内的各项功能。

4.2 小程序首页

当用户登录小程序后，他们就会进入到首页，这里有许多功能可供用户使用。用户可以通过柱状图查看当前时间的座位信息，并且图书推荐栏会实时向用户推荐当前最受欢迎的三本图书，以提供用户最佳的使用体验。

4.3 座位预约功能

通过座位预约功能，用户可以轻松实现对图书馆座位的在线预约。该功能为用户提供了一个便捷的平台，使他们能够在任何时间、任何地点进行座位预约，避免了现场排队等待的麻烦。用户只需在系统中选择所需的座位、预约时间和时长，系统将自动为其保留所选座位。用户还可以随时查看座位的预约情况，了解座位的实时状态，以便做出更合理的预约选择。座位预约功能的实现，不仅提高了座位的使用效率，也提升了用户的体验满意度。

4.4 图书功能

通过该功能用户可以查看书本的详细情况，包括位置，描述等信息。用户只需简单操作，便可获取所需信息，极大地提升了借阅体验，使得图书的查找与选择变得更为便捷与高效。同时，这也为用户提供了更个性化的阅读推荐，助力他们在书海中快速找到适合自己的读物。

4.5 我的座位功能

通过该功能，用户可以方便的查看自己在图书馆的预约座位信息。无论是签到，还是续约，都能一键完成。这一功能不仅提供了座位预约的便捷性，还确保了座位资源的合理分配与高效利用。用户可以根据自己的需求，提前预约座位，避免到图书馆后找不到座位的尴尬情况。

结语

随着中国经济快速增长，消费者对移动设备的需求不断增加，图书馆座位智能管理系统应运而生。该系统通过微信小程序客户端和网页管理端，为用户提供便捷的座位预约服务。在设计过程中，注重功能全面性和用户友好界面，同时研究座位与预约信息以提升可操作性。系统基于 SSM 架构，使用 Java 语言、MySQL 数据库和 Vue 框架，实现了用户管理、图书管理、座位管理等主要功能。该系统以简单的接口和易于操作的特点，显著提升了用户体验，满足了消费者对移动服务的需求。

参考文献

[1] 张春亮.计算机应用的现状与计算机的发展趋势[J].电脑迷,2017,10(1):1-4.
[2] 李红娣.图书馆移动座位管理信息系统设计与实现[M].江西财经大学,2017.
[3] 唐绍富.基于.NET 技术的怀化职院图书管理系统的设计与实现[M].电子科技大学.2009.
[4] 蒋谢芳,马骏,王长浩,高健.智慧图书馆座位管理系统设计与实现[J]数字技术与应用,2019(6):158-159.
[5] 黄元培.网络数据库安全隐患及防范措施解析[J].无线互联科技,2020(9):28-29.
[6]Yongbin Zhang; Ronghua Liang; Yuansheng Qi; Hejie Chen; Bingran Li.Yongbin Zhang; Ronghua Liang; Yuansheng Qi;Hejie Chen;Bingran Li[J].2021 7th Annual International Conference on Network and Information Systems for Computers (ICNISC),2021(6):23-25.
[7] 黄元培.网络数据库安全隐患及防范措施解析[J].无线互联科技,2020(9):28-29.

作者简介：潘俊璋(2002—),男，汉族，河北迁安人，硕士在读，研究方向：人工智能；
通信作者：席亚军（1981—），男，汉族，河北平泉人，本科，图书管理员，研究方向：图书馆信息化。
基金项目：CALIS 全国农学文献信息中心研究项目+图书馆座位智能管理系统的设计与实现+（2024051）。

3. 论文：

Liu et al. A cotton organ segmentation method with phenotypic measurements from a point cloud using a transformer[J]. Plant Methods, 2025: 21:37.（中科院二区， IF=4.2）

Liu *et al. Plant Methods* (2025) 21:37
https://doi.org/10.1186/s13007-025-01357-w

Plant Methods

METHODOLOGY Open Access

A cotton organ segmentation method with phenotypic measurements from a point cloud using a transformer

Fu-Yong Liu[1], Hui Geng[2], Lin-Yuan Shang[2], Chun-Jing Si[2,3*] and Shi-Quan Shen[1*]

Abstract

Cotton phenomics plays a crucial role in understanding and managing the growth and development of cotton plants. The segmentation of point clouds, a process that underpins the measurement of plant organ structures through 3D point clouds, is necessary for obtaining precise phenotypic parameters. This study proposes a cotton point cloud organ semantic segmentation method named TPointNetPlus, which combines PointNet++ and Transformer algorithms. Firstly, a dedicated point cloud dataset for cotton plants is constructed using multi-view images. Secondly, the attention module Transformer is introduced into the PointNet++ model to increase the accuracy of feature extraction. Finally, organ-level cotton plant point cloud segmentation is performed using the HDBSCAN algorithm, successfully segmenting cotton leaves, bolls, and branches from the entire plant, and obtaining their phenotypic feature parameters. The research results indicate that the TPointNetPlus model achieved a high accuracy of 98.39% in leaf semantic segmentation. The correlation coefficients between the measured values of four phenotypic parameters (plant height, leaf area, and boll volume) ranged from 0.95 to 0.97, demonstrating the accurate predictive capability of the model for these key traits. The proposed method, which enables automated data analysis from a plant's 3D point cloud to phenotypic parameters, provides a reliable reference for in-depth studies of plant phenotypes.

Keywords Cotton phenotype, Point cloud, TPointNetPlus, Attention mechanism, Semantic segmentation

Introduction

The climate and soil conditions in Xinjiang provide an ideal environment for the growth of cotton, making it one of the most significant cotton-producing regions in China [1]. However, the prosperity of the cotton industry is directly linked to the livelihoods of local farmers and the economic development of the region [2]. Research on cotton phenotypes allows for a more accurate understanding of the plant's physiological status, adaptability, and response to environmental changes [3]. This approach is crucial for increasing cotton yield, improving quality, and cultivating more resilient varieties. Precise measurements of various cotton traits, such as plant height, leaf morphology, and boll size, enable a better assessment of the growth conditions of plants under different varieties or treatments, providing targeted recommendations for breeding and cultivation [4]. However, traditional methods often rely on manual measurements and observations, leading to subjectivity and a high workload [5]. Two-dimensional image measurement methods, including pixel-based analysis, feature extraction and matching, geometric shape fitting, and visual

*Correspondence:
Chun-Jing Si
smalls1002@163.com
Shi-Quan Shen
ssquan123@163.com
[1] College of Information Science and Engineering, Xinjiang University of Science and Technology, Korla 841000, China
[2] College of Information Engineering, Tarim University, Alaer 843300, China
[3] Key Laboratory of Tarim Oasis Agriculture (Tarim University), Ministry of Education, Alaer 843300, China

measurement techniques, may encounter challenges such as image quality, deformation, feature extraction stability, scale, and computational complexity [6].

The use of three-dimensional point cloud technology for obtaining plant organ parameters offers several advantages, including precise structural information, non-invasive measurement, comprehensive data retrieval, adaptability to complex environments, and automation for efficiency. This method serves as a valuable tool for plant phenotyping research and agricultural production [7–9]. Various technologies used for point cloud acquisition include laser scanning (Lidar), structured light, time-of-flight (ToF) cameras, stereo vision, multi-view photogrammetry, panoramic photography, and sonar scanning. These technologies utilize sensors like lasers, light patterns, cameras, and sound waves to measure surface attributes and generate three-dimensional coordinate data in the form of point clouds [10–12]. The generation of such data has facilitated the use of deep learning for point cloud segmentation [13, 14].

Plant point cloud segmentation and measurement technology based on deep learning combines deep learning algorithms with point cloud processing, enabling accurate segmentation of plant structures and precise measurement of phenotypic parameters. By utilizing deep learning algorithms such as convolutional neural networks, this technology can learn and extract complex plant features, providing robust support for point cloud segmentation and feature extraction. Techniques such as PointNet++ have been applied to plant point cloud segmentation, demonstrating their potential in identifying and analyzing different plant parts, such as leaves, stems, and fruits [15–17]. By segmenting plant point clouds into these parts, researchers can finely measure and analyze the phenotypic parameters of each part, such as plant height and leaf area [18, 19]. This technology has found widespread applications in agricultural research, plant science, and agricultural production, offering a new approach to understanding plant growth, adaptability, and environmental response [20]. The study provides a scientific basis for breeding and agricultural management.

However, despite the significant advancements made by PointNet++ in local feature extraction, its application in plant point cloud segmentation and measurement still has limitations. Specifically, PointNet++ may not fully capture all the fine local features when dealing with the complex structures and diverse morphological characteristics of plants. Its hierarchical structure has limitations in capturing large-scale and long-distance dependencies, potentially leading to information loss. Additionally, PointNet++ shows insufficient robustness to noise and occlusions commonly encountered in practical agricultural scenarios, affecting the accuracy of segmentation and measurement.

This study presents an effective approach for addressing the difficulties in extracting plant phenotypes from 3D point clouds. This approach comprises acquiring plant point clouds, creating appropriate datasets, and partitioning plant point clouds. Firstly, a dedicated point cloud dataset for cotton plants is constructed using multi-view images. Secondly, the attention module Transformer is introduced into the PointNet++ model to increase the accuracy of feature extraction. Finally, organ-level cotton plant point cloud segmentation is performed using the HDBSCAN algorithm, successfully segmented cotton leaves, bolls, and branches from the entire plant, and obtained their phenotypic feature parameters. Our primary contributions can be outlined as follows:

- We constructed a high-precision, dense point cloud dataset of cotton plants using Structure from Motion (SfM) 3D structural reconstruction. The dataset consists of over 724 high-quality partial and complete point clouds, documenting the growth of the plants over time. Each cotton plant is represented by a 3D point cloud containing 40,960 points.
- The integration of the Transformer module into the PointNet++ network was aimed at improving the accuracy of cotton plant point cloud organ segmentation.
- Leaves, bolls, and branches obtained after the segmentation of the instances have a correlation coefficient greater than 0.9 between the predicted values and the true values obtained from the point cloud computation, which can be applied to the actual production.

Related works

Plant point cloud segmentation and measurement

Despite the significant progress in plant point cloud segmentation and measurement technology based on deep learning in plant phenotyping research, some challenges still exist. Meng et al. designed a Vv-Net [21] to voxelize point clouds. Since transforming point clouds into images or voxels cannot effectively utilize the spatial features of point clouds and increase the data processing cost and structured noise to varying degrees, Xu et al. [22] constructed a convolution kernel with a dynamic convolutional weight matrix and proposed a position adaptive convolution (PAConv), for which the weight coefficients of the matrix can be obtained by the fractional network adaptively learning the relative positional relationships of the points. To address the problem that point clouds need to be transformed into images or voxels first when

using 2D convolution or 3D convolution in point cloud semantic segmentation, in addition to designing point convolution with point clouds as input, GNNs can be construct to build a special graph structure about the point clouds [23–25], then graph convolution can be used to explore the neighboring information of each point to better utilize the spatial features of the point clouds and to improve the segmentation accuracy. The dynamic graph convolutional neural network (DGCNN) designed by Wang et al. [23] takes the input N points as the centers, computes the respective K nearest neighbors layer by layer to dynamically construct the local neighborhood graph, and then uses edge convolution to compute the edge features between the center point and its nearest neighbors. However, the fixed size of the edge features prevents the model from performing well at different scales and input points. Therefore, we have constructed a point cloud semantic segmentation network based on the deep learning network PointNet++. This has solved the current problem of insufficient accuracy in plant organ segmentation.

Transformer

In the field of computer vision, transformer model processes global information mainly through an attention mechanism, and the core idea is to compute an output representation for each location by focusing on different parts of the input sequence [26–30]. DGANet [31] further differentiates each edge of the constructed local graph by integrating a dilated graph attention module implemented by an offset attention mechanism to better learn the edge features. PAN [32] is based on a novel local attention edge convolution layer and a point-by-point spatial attention module. Although the attention mechanism allows the model to filter and learn the most important information [33, 34], researchers often need to expend much effort designing special attention modules for different tasks, such as channel attention and spatial attention [35], and the computational complexity of different attention modules is different, which does not support parallel computation. By introducing the multi-head attention mechanism, the transformer can focus on different aspects of the input sequence at the same time, thus capturing the global information effectively. In addition, the Transformer model can be further enhanced by stacking multiple Transformer layers to further enhance its modeling capabilities. This paper introduces the integration of the Transformer module into the PointNet++ network, resulting in a novel point cloud transformer structure tailored for enhancing segmentation accuracy in 3D point clouds within the PointNet++ network.

The self-attention mechanism of Transformer is able to capture global features and long-range dependencies in point cloud data more effectively. This enhances the ability to understand and capture details of complex plant structures, especially when dealing with subtle and complex features. Additionally, Transformer exhibits greater robustness in the face of noise and occlusion, improving the overall understanding and segmentation accuracy of point cloud data through information balancing and optimization on a global scale. Therefore, adding Transformer to the PointNet++ network encoder can improve the accuracy of cotton plant organ segmentation.

Materials and methods

Overview

The proposed method for cotton organ segmentation and measurement is illustrated in the flowchart in Fig. 1, presenting a comprehensive framework. This method is divided into four fundamental components: image data acquisition, generation of a cotton plant point cloud, segmentation of the cotton organ point cloud, and extraction of organ phenotypic parameters.

Cotton3D dataset

Data acquisition

The Xinjiang cotton experiment was conducted in 2021 at the East District of Tarim University in Alar city, located in the Aral Reclamation Area of Xinjiang. This area is situated at the northern edge of the Taklamakan Desert and at the confluence of the Aksu, Hotan, and Yarkant Rivers in the upper reaches of the Tarim River.

In this experiment, the later cotton plants were tall, with a mean plant height of approximately 1.1 m, and a mean distance between the longest leaves on both sides of the plant at the same level as the body, between 50 and 60 cm. Cotton was placed on the centre table. Due to the lack of light in the laboratory, three additional photographic lights were added to supplement the light, each placed 120 degrees apart horizontally, with the lights positioned 2 m above the ground. The actual shooting environment was arranged as shown in Fig. 2. Because the plant was too large, it was not suitable to use a motorized turntable to carry out the rotational shooting, because the plant itself was difficult to fix, resulting in shaking to produce a large amount of noise or failure to align the image. Therefore, this experiment used purely manual imaging to obtain images. When photographing the cotton plants, the distance between the camera and the cotton plant was approximately 30 cm. The camera was used to take pictures at elevated, flat, and overhead shooting angles, and one picture was taken at the same height at intervals of approximately 6 degrees, depending on the visual angle of the camera. Approximately

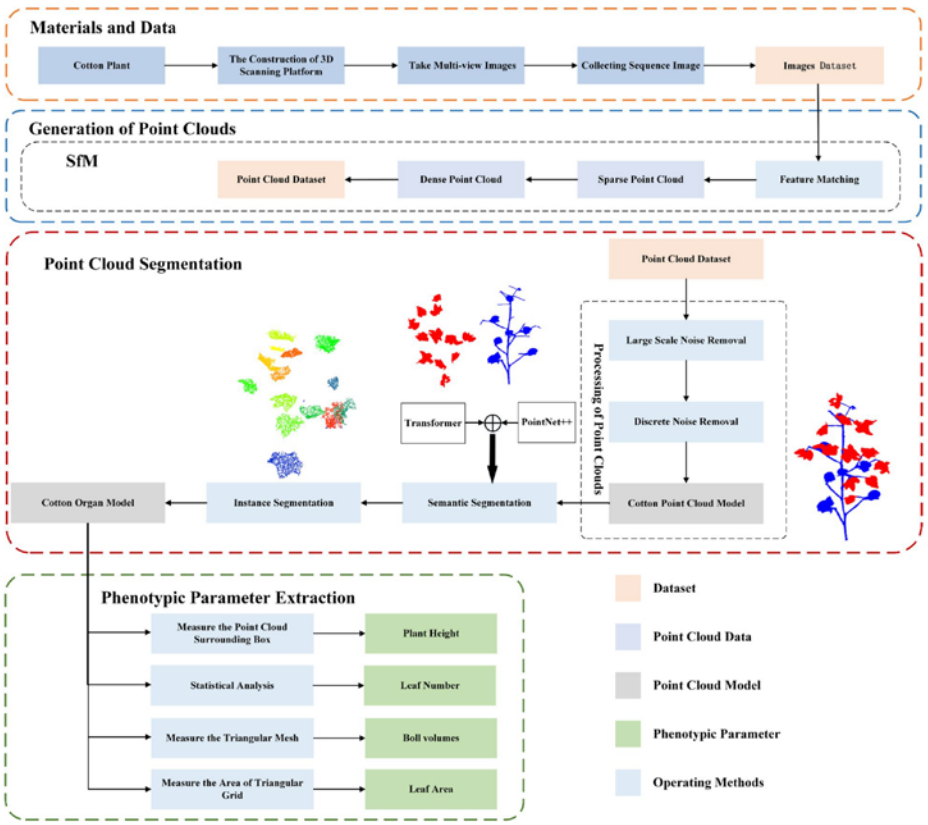


Fig. 1 The flowchart of our method. Firstly, image data is acquired using advanced imaging techniques to capture detailed information of the cotton plant. This data is then used to generate a point cloud that represents the three-dimensional structure of the cotton plant. The next step involves segmenting the point cloud to isolate individual cotton organs. Finally, phenotypic parameters of these organs

60 pictures were taken at each shooting angle, totaling approximately 180 pictures per cotton plant.

Point cloud preprocessing

3D reconstruction of cotton plants Utilizing RealityCapture's Motion Recovery Structure technology, objects of the same target are photographed multiple times from different angles. The photos are then filtered to exclude blurred, out-of-focus, and dissimilar photos to avoid compromising the accuracy of the reconstruction. Next, the images are imported into the software to generate spatial point cloud pixel values using a dense reconstruction algorithm. Multiple stereo matching algorithms are then used to reconstruct the images. Once the reconstruction is complete, camera calibration parameters are applied to eliminate lens aberrations, automatically remove outliers, and concatenate the point cloud to form a triangular mesh file for post-measurement.

Point cloud normalization Point cloud normalization is the process of scaling point cloud data to a specified range along three coordinate axes (X, Y, Z). Typically, point cloud normalization scales the point cloud into a unit cube with the center point of the cube as the origin $(0, 0, 0)$ and the length of its side as 1. The purpose of point cloud normalization is to map different sizes

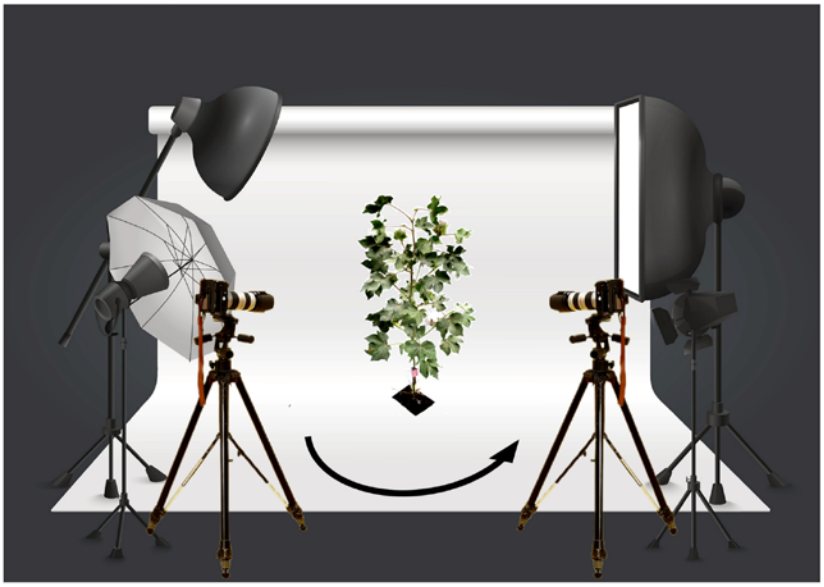


Fig. 2 Image data acquisition platform

of point cloud data into the same scale space, which is beneficial for subsequent data processing and analysis.

Data labels The data labels for 3D reconstruction of cotton are key information used to identify and describe the point cloud data, which helps to distinguish between different objects, parts, or features. These labels can include object labels, part labels, and feature labels. In this paper, when using CloudCompare software to label the reconstructed 3D point cloud data of cotton plants, the following steps can be taken opening the point cloud data file of cotton plants, and classifying and labelling the leaves, bolls, and branches according to demand. All points in the point cloud of leaf organs were labeled 1, those in the point cloud of boll organs should be labeled 2, and the point cloud of branch organs should be labeled 3, with the rest given a value of 0. In CloudCompare, the point cloud data are labeled interactively through the view interface.

Cotton3D dataset

The Cotton3D dataset is a comprehensive collection of 3D point cloud data that captures the growth cycle of cotton plants. The original cotton plant point cloud data was acquired using a camera and SfM techniques, and therefore the dataset exhibits greater morphological irregularities and discontinuities in surface features. The dataset consists of more than 724 high-quality partial and complete point clouds detailing plant growth over time. Each cotton plant is represented by a three-dimensional point cloud containing 40,960 points. The dataset was further expanded with data enhancement methods such as random rotation (angle_sigma = 0.05, angle_clip = 0.1), random noise (sigma = 0.01, clip = 0.05), disrupted point clouds, and random scaling (scale_low = 0.95, scale_high = 1.05). The data with better plant integrity was finally selected to be brought into the network to complete the segmentation task. The Cotton3D dataset used for the segmentation task is shown in Table 1.

Table 1 The settings for the training dataset and test dataset

	Number of training point clouds	Number of testing point clouds	Points (average)
Number of plants	288	36	40,960
Number of leaves	4648	576	1592
Number of bolls	1568	196	1110
Number of stems	288	36	9248

Transformer

Transformer in point cloud processing captures global relationships through its self-attention mechanism, thereby enhancing the richness and accuracy of feature representation. The transformer architecture [36] is as follows (Fig. 3): First, the point cloud data with input dimensions $\{N, K, d+C\}$, where N is the number of points, K is the number of neighbors per point, and $d+C$ is the feature dimension, passes through a Multi-Layer Perceptron (MLP) to extract initial features, denoted as $q_j^{i-1}, k_j^{i-1}, v_j^{i-1} = MLP(X)$, where $X \in \mathbb{R}^{N\times K\times(d+C)}$. Next, position encoding P is added to the initial features to incorporate positional information, resulting in $F = v_j^{i-1} + P$. The processed features then undergo a series of element-wise operations: element-wise subtraction $S = q_j^{i-1} - k_j^{i-1}$, element-wise addition $A = S + P$, and element-wise multiplication $M = A \odot F$, where $\odot$ denotes element-wise multiplication. Finally, the results of these operations are merged and processed by another MLP to produce the final output $F_{out} = MLP(M)$, which has the same dimensions as the input $\{N, K, d+C\}$. This architecture enhances the feature representation of the point cloud, enabling the Transformer to more effectively process point cloud data and improve processing performance.

The advantages of the transformer model include its parallel processing capability and its ability to model long sequences. Moreover, this approach can better capture the semantic information in the input sequence because the self-attention mechanism can focus on all positions in the sequence at the same time. However, compared with those of traditional RNN and CNN models, the computational complexity of the transformer model is greater, requiring a large amount of computational resources and training data. In recent years, the visual transformer model has become one of the mainstream models in the field of computer vision, and its application scenarios and model structure have been continuously extended and improved. For example, models such as DETR [30] and ViT [37] are based on the improved version of the visual transformer, and these models have undergone many related optimizations based on 2D vision tasks. For example, it is unrealistic to input all the pixels of a 2D image into the fully connected layer for computation at the same time, which greatly exceeds the capacity of the fully connected layer. In VIT, the network first slices the input image into multiple 16×16 patches, and then the patches are used as the smallest unit for inputting the multi-attention mechanism, which significantly reduces the difficulty of the computation, and achieves excellent performance.

Instance segmentation of point clouds

Semantic segmentation

The improved PointNet++ network is used for semantic segmentation of cotton plant point clouds. PointNet++ [38] achieves better descriptions of local features and overall features than PointNet dose. PointNet fundamentally learns the information of each point to obtain the spatial encoding, after which all the obtained features of all the points are aggregated together to become the point cloud's global features. PointNet, however, is unable to acquire structural information between the points of the point cloud or more compatible local features due to its network structure. Therefore, constructing local features of the point cloud is a crucial part of the network design, as only when the neural network is provided with enough receptive fields and receptive domains, can it acquire better features. PointNet++ uses the spatial information between points as a criterion for dividing the point cloud. This method divides the point cloud into one overlapping local space, and then spatially

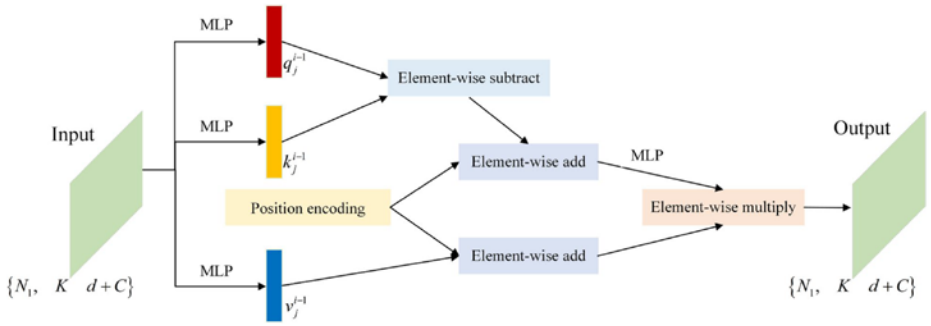


Fig. 3 The detailed structure of transformer

encodes the points in the local space to obtain the local features of each region. The design of local features takes into account the structural and geometric information of the point cloud and can improve the semantic segmentation of the point cloud. In this study, the incorporation of Transformer into the feature extraction component of PointNet++ addresses the limitations of the original model and compensates for the lack of localized features and the ability to capture global relationships in point cloud data, as shown in Fig. 4.

In the sampling process, the distance metric between points is used as a criterion by using the Euclidean distance, which is calculated in Eq. (1) for points $p_i(x_i, y_i, z_i)$ and $p_j(x_j, y_j, z_j)$.

$$D\left(p_i, p_j\right) = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2} \tag{1}$$

The PointNet layer, on the other hand, utilizes a simple PointNet structure to form a local spatial feature extraction module. The function of the underlying PointNet network is to map an unordered set of point clouds $\{x_1, x_2, \cdots, x_n\}$ onto a single vector using a function that behaves as follows.

$$f(x_1, x_2, \cdots, x_n) = \gamma\left(\underset{i=1,\cdots,n}{MAX}\left\{h(x_i)\right\}\right) \tag{2}$$

where the networks γ and h usually behave as multi-layer perceptrons, and h corresponds to the encoding of the local spatial information of the point cloud. Through the PointNet layer, the information of the points in the local space is finally aggregated into a one-dimensional vector.

The realization process involves performing point feature propagation through distance-based interpolation, and aggregating the carry features of the corresponding points in the corresponding coding layer through cross-layer skip link concatenation. When carrying out point feature propagation from the N_l layer to the N_{l-1} layer, assuming that we want to obtain the features of point A in the N_{l-1} layer, we first use the KNN interpolation approach to find the three nearest points of point A in the N_l layer, and carry out the weighted summation, and the corresponding $p = 2$ and $k = 3$ in Eq. (3). Then we use the weighted features obtained and the corresponding set of points in the encoding process, the SA abstract layer, to aggregate their corresponding points in the corresponding coding layer through the cross-layer Skip Link Concatenation obtained during the encoding process are combined through cross-layer jump links, and the connected combined features are aggregated through a single PointNet layer structure. Different point cloud upsampling steps are performed by this feature aggregation approach until the original point cloud size is restored to the original point cloud size.

$$f^{(j)}(x) = \frac{\sum_{i=1}^{k} w_i(x) f_i^{(j)}}{\sum_{i=1}^{k} w_i(x)} where w_i(x) = \frac{1}{d(x, x_i)^p}, j = 1, \cdots, C \tag{3}$$

Clustering algorithm

After semantic recognition using TPointNetPlus, this study utilized the HDBSCAN (Hierarchical Density -Based Spatial Clustering of Applications with Noise) [39] algorithm to achieve organ-level instance segmentation of cotton (Fig. 5). HDBSCAN is a clustering method that utilizes DBSCAN and hierarchical clustering, both of which are well-known. It is a density-based spatial data clustering technique that determines the reachable distances between neighboring points and core points to construct an inter-arrival graph. The last step is to use hierarchical clustering and clustering tree compression to form the clusters.

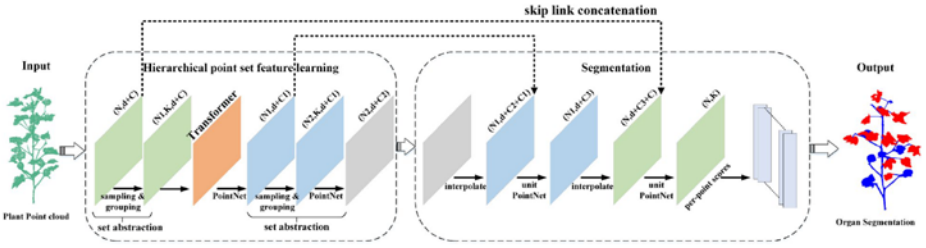


Fig. 4 The network structure of TPointNetPlus semantic segmentation. The left part is the coding part of the network, which gradually downsamples the point cloud and mainly realizes the local feature space coding of the point cloud, while the right half is the decoding part of the network, which upsamples each point in the point cloud and gradually restores the structure to the original point cloud, and simultaneously realizes the feature information aggregation of each point in the point cloud. Finally, each point is semantically discriminated by the fully connected layer

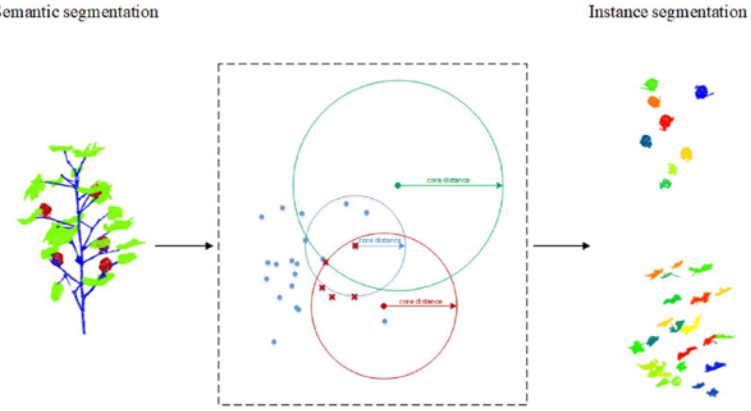


Fig. 5 The mutual reachability distance of HDBSCAN

HDBSCAN does not require users to pre-determine the number of clusters or the distance threshold around cluster points. The reachable distance design can deal with clusters with different density distances, constructing a hierarchical structure based on density clustering, thereby facilitating more efficient extraction of discontinuities. The mutual reachability distance between two points is defined as Eq. (4).

$$d_k(p,q) = \max\{c_k(p), c_k(q), d(p,q)\} \tag{4}$$

where $d(p,q)$ denotes the distance between points p, and q and the core distance $c_k(p) = d(p, N^k(p))$ denotes the distance between core point p and the kth neighboring point.

Phenotype parameter extraction

Cotton leaf area The area of a leaf is often calculated by converting the point cloud data into a grid through the process of triangulation. Let there be a point cloud with N points, each represented by coordinates (x_i, y_i, z_i) for $i = 1,2,...,N$. Let M be the number of triangles in the mesh. For each triangle j, defined by three vertices (x_{j1}, y_{j1}, z_{j1}), (x_{j2}, y_{j2}, z_{j2}), (x_{j3}, y_{j3}, z_{j3}), the lengths of the three sides of the triangle were computed as $a_j = ||P_{j2} - P_{j1}||$, $b_j = ||P_{j3} - P_{j2}||$, $c_j = ||P_{j1} - P_{j3}||$, calculate the semi-perimeter, s_j was calculated as $s_j = \frac{a_j+b_j+c_j}{2}$, Heron's formula was applied to compute the triangle area $A_j = \sqrt{s_j \cdot (s_j - a_j) \cdot (s_j - b_j) \cdot (s_j - c_j)}$, and the areas of all the triangles were summed to obtain the total surface area $S = \sum_{j=1}^{M} A_j$.

Cotton plant height The x, y, and z coordinates of all the point clouds from the segmented cotton plant stems were extracted, normalized, and processed to obtain a normalized coordinate matrix. The elements of each row in the matrix were summed, compressed into a single column, and subsequently subjected to square root summation to obtain the centroid of the organ. Subsequently, the distance from the organ's centroid to the vertex was calculated, allowing the deduction of the reference point cloud's edge length. This edge length was subsequently used to determine the scale factor. By multiplying the differences between the maximum and minimum values in the x, y, and z directions by the scale factor, estimated values for various organs of the cotton plant were obtained.

$$H_{stem} = \frac{F_{reference}}{E_{reference}} \times F_{stem} \times S \tag{5}$$

where H_{stem} is the cotton organ estimate, $F_{reference}$ is the actual value of the reference, $E_{reference}$ is the projected value of the reference, and F_{stem} is the actual value of the cotton organ. S is the proportionality adjustment factor, which was experimentally verified by setting 0.94.

Boll volumes To compute the volume of a point cloud, a common method involves voxelization. In this process, the continuous space occupied by the point cloud is discretized into small cubic elements known as voxels. Let there be N points in a point cloud, with the same coordinates (x_i, y_i, z_i), where $i = 1,2,\ldots,N$. The voxel size is denoted as $\Delta x \times \Delta y \times \Delta z$. By mapping each point to discrete voxel coordinates (v_x, v_y, v_z), which are calculated as

$v_x = \lfloor \frac{x_i}{\Delta x} \rfloor$, $v_y = \left\lfloor \frac{y_i}{\Delta y} \right\rfloor$, $v_z = \lfloor \frac{z_i}{\Delta z} \rfloor$, the point cloud is voxelized. The volume of each voxel is $\Delta x \times \Delta y \times \Delta z$, and the total volume is obtained by multiplying the number of voxels concerning the volume of each voxel. This voxel-based approach provides an estimate of the point cloud volume, offering a discrete representation suitable for computational analysis.

Evaluation metric

Evaluation metrics for point cloud segmentation

The evaluation metrics for the cotton organ point cloud segmentation results include the accuracy, intersection and concatenation ratio (IoU) and cross-entropy loss function. In calculating the accuracy rate, the predicted values (p_i) generated by the TPointNetPlus model were compared with the actual labels (y_i) of the point cloud to evaluate the degree of consistency between them. The IoU metric evaluates the degree of overlap between the predicted label set and the actual label set, whereas the cross-entropy loss function quantifies the difference between the predicted and actual values [40]. To address the category imbalance problem, a weighted cross-entropy loss function was used to assign weights (w_i) to each category based on the number of point clouds in each category. This weighting mechanism aims to prioritize categories with fewer point clouds, ensuring a fair assessment of the model's performance across all categories [41]. In addition to the TPointnetPlus module, factors such as the architecture of the network model, the quality and diversity of the dataset, and the design of the training process also have impacts on the accuracy of the model.

$$\mathrm{Acc} = \frac{1}{m}\sum_{i=1}^{m} f_i, f_i = \begin{cases} 1 \; y_i = p_i, \\ 2 \; y_i \neq p_i \end{cases} \tag{6}$$

$$\mathrm{Loss} = \sum_{\mathrm{i}=1}^{\mathrm{m}} (\mathrm{y_i}\log(\mathrm{p_i}) + (1-\mathrm{y_i})\log(1-\mathrm{p_i})) \tag{7}$$

$$\begin{aligned} \mathrm{IoU} &= \frac{\text{Area of overlap}}{\text{Area of union}} \\ &= \frac{\text{Area}(\text{prediction} \cap \text{target})}{\text{Area}(\text{prediction} \cup \text{target})} \\ &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} \end{aligned} \tag{8}$$

$$\mathrm{w}(\mathrm{p},\mathrm{y}) = \sum_{\mathrm{i}=1}^{\mathrm{C}} \mathrm{w_i p_i} \log(\mathrm{y_i}) \tag{9}$$

Evaluation metrics for measured cotton organs

The true values of the measured cotton organ lengths, widths and heights were compared with the model-calculated estimates of the cotton organs, and accuracy was assessed by the correlation coefficient (R), root mean square error ($RMSE$), and margin of error (δ). The predicted values of cotton plant organ phenotypic data $\widehat{\mathrm{Y}} : \left\{\widehat{\mathrm{Y}}_1, \widehat{\mathrm{Y}}_2, \cdots, \widehat{\mathrm{Y}}_\mathrm{n}\right\}$, and the true values of the cotton plant organ phenotypic data $\mathrm{Y} : \{\mathrm{Y}_1, \mathrm{Y}_2, \cdots, \mathrm{Y}_\mathrm{n}\}$ were obtained. The correlation coefficient is calculated as shown in Eqs. 10–13.

$$\mathrm{E}\left(\widehat{\mathrm{Y}}\right) = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \widehat{\mathrm{Y}}_\mathrm{i}}{\mathrm{n}}, \; \mathrm{E(Y)} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{Y_i}}{\mathrm{n}} \tag{10}$$

$$\mathrm{Cov}\left(\widehat{\mathrm{Y}}, \mathrm{Y}\right) = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \left(\widehat{\mathrm{Y}}_\mathrm{i} - \mathrm{E}\left(\widehat{\mathrm{Y}}\right)\right)(\mathrm{Y_i} - \mathrm{E(Y)})}{\mathrm{n}} \tag{11}$$

$$\sigma_{\widehat{\mathrm{Y}}} = \sqrt{\frac{\sum_{i=1}^{n} \left(\widehat{\mathrm{Y}}_i - E\left(\widehat{\mathrm{Y}}\right)\right)^2}{n}}, \sigma_Y = \sqrt{\frac{\sum_{i=1}^{n} (Y_i - E(Y))^2}{n}} \tag{12}$$

$$R = \frac{\mathrm{Cov}\left(\widehat{\mathrm{Y}}, \mathrm{Y}\right)}{\sigma_{\widehat{\mathrm{Y}}}\sigma_Y} = \frac{\sum_{i=1}^{n} \frac{\left(\widehat{\mathrm{Y}}_i - E\left(\widehat{\mathrm{Y}}\right)\right)}{\sigma_{\widehat{\mathrm{Y}}}} \frac{(Y_i - E(Y))}{\sigma_Y}}{n} \tag{13}$$

where $\mathrm{E}\left(\widehat{\mathrm{Y}}\right)$ and $\mathrm{E(Y)}$ are the overall means of $\widehat{Y}$, and Y respectively. $\mathrm{Cov}\left(\widehat{\mathrm{Y}}, \mathrm{Y}\right)$ is the population covariance. $\sigma_{\widehat{\mathrm{Y}}}$ and σ_Y are the standard deviations of $\widehat{Y}$ and Y respectively. R is the correlation coefficient, where the higher the value is the more accurate the prediction.

$RMSE$ is a commonly used measure of the difference between the predicted and actual observed values of a model to assess how well the model fits the given data. The smaller this value is the more valid the model. $RMSE$ is calculated as shown in Eq. 14.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=0}^{n} \left(\mathrm{Y_i} - \widehat{\mathrm{Y}}_\mathrm{i}\right)^2} \tag{14}$$

where n is the number of samples, and $\mathrm{Y_I}$ and $\widehat{\mathrm{Y}}_\mathrm{I}$ are the organ estimates and actual organ measurements, respectively.

$$\delta = \frac{\Delta}{L} \times 100\% \tag{15}$$

where Δ is the absolute value of the algorithm's estimate subtracted from the actual measurement and L is the actual measurement of the trait.

Table 2 Hardware configuration and operating environment

Hardware	Configure	Environment	Version
CPU	Intel(R)Xeon(R)CPUE5-2678v3@2.50 GHz	OS Version	Ubuntu 18.04
GPU	GeForce RTX 2080Ti 10G	Framework	Pytorch 1.11
RAM	16 GB DDR4-3200 MHz	CUDA	CUDA11.6 Cudann8.1.1
Hard Disk	BC711 512 GB NVME SSD	Python	3.7

Table 3 Experimental results of hyperparametric network training

hyper-parameters	Strategy	Accuracy (%)
Optimizer	SGD	95.3
	Adam	95.3
Learning rate	0.25	92.6
	0.5	95.3
	0.75	95.1
Decay rate	0.01	91.3
	0.0001	95.3
Epoch	100	94.8
	200	95.6
	500	95.8

Results

Detailed settings

The training environment for this experiment is Intel (R) Xeon (R) CPU E5-2678v3 at 2.50 GHz; NVIDIA GeForce RTX 2080Ti 10G GPU; and the operating system is Ubuntu18.04. The deep learning environment is CUDA11.6, cuDNN8.1.1, Pytorch1.11, Python3.7. Implementation details, listed in Table 2.

In order to maximize the accuracy of the modeling results, hyperparametric experiments were conducted in this study, as shown in Table 3. The optimizer ADAM trained TPointNetPlus with an initial learning rate of 0.5, a batch size of 6, and an iterative calendar element count of 500. All networks were trained end-to-end using stochastic gradient descent. These network parameters trained the model with the highest test accuracy.

Results of the improved network

The training process

After training the TPointNetPlus network, this study tested the point clouds of cotton plants on the network for organ segmentation. The results, depicted in Fig. 6a, showed that TPointNetPlus had the highest accuracy. Compared to PointNet, both TPointNetPlus and PointNet++ had higher accuracy, with PointNet's accuracy being limited to less than 80% due to its inability to capture localized details. TPointNetPlus had an accuracy more than 5% greater than PointNet++. The performance of the network was evaluated by analysing the loss function values during the training process in Fig. 6b. Initially, PointNet had highly unstable values with large fluctuations, while TPointNetPlus had some regions with greater fluctuations in loss than PointNet++. However, from $Epoch = 100$ onwards, TPointNetPlus began to level off, while PointNet++ and PointNet still exhibited large fluctuations. By $Epoch = 500$, the actual loss values had decreased to less than 0.5 for PointNet, less than 0.4 for PointNet++, and below 0.2 for TPointNetPlus. After training, the average loss for TPointNetPlus was approximately 0.08, whereas the average loss for PointNet++ which was approximately 0.14.

Quantitative comparison

Similarly, the data in the test set were input into the PointNet and PointNet++ networks. The experimental results showed that the TPointNetPlus network

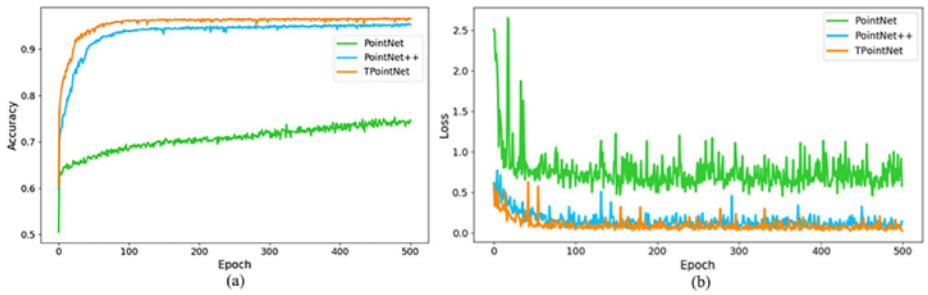


Fig. 6 Visualization of three network training processes. **a** Accuracy values during the training of the networks; **b** The loss function values during the training of the networks

outperformed the PointNet and PointNet++ networks in terms of several evaluation metrics, such as accuracy, F1 score, and mIoU, for the performance of leaf, boll, branch, and overall segmentation of the cotton plant. As shown in Table 4, the PointNet++ networks had the lowest accuracy compared to the other networks. In terms of accuracy, TPointNetPlus achieved the greatest improvement in the accuracy of individual organ segmentation. Compared to the other networks (PointNet and PointNet++), TPointNetPlus improved by 18.65%, 18.12%, 13.14%, and 5% for leaves, bolls, branches, and overall, respectively. This is because leaves have a larger area with distinctive overall features, whereas bolls and branches are smaller and more dispersed. Therefore, the segmentation effect on cotton leaves was more than 80%, while the accuracy of boll and branch segmentation was less than 40%. In terms of overall segmentation accuracy, there was not much difference between TPointNetPlus and PointNet++, but there was a significant difference in the segmentation accuracy scores for each organ. This indicates that TPointNetPlus is more effective than PointNet++ in local detail feature extraction.

According to the F1 score, the performance of leaves, bolls, branches, and overall performance exceeded 90%, with the values for leaves exceeding 99%. This indicates that TPointNetPlus has achieved a good balance of accuracy and recall. Due to the narrowly dispersed feature area of the branches, the mIoU value of TPointNetPlus is only 80.37%. However, the value of the PointNet network is only 19.06%, which is enough to prove the effectiveness of TPointNetPlus in local feature extraction. This finding suggested that the model can accurately capture the location and shape of the target during segmentation.

Table 4 Segmentation results from three deep learning networks on the test set

Metric	Organ	PointNet	PointNet++	TPointNetPlus
Accuracy (%)	Cotton leaf	80.04	92.12	**98.39**
	Cotton boll	30	86.63	**93.59**
	Cotton stem	37.87	72.12	**83.03**
	Overall	61.4	90.53	**95.3**
F1 score (%)	Cotton leaf	88.57	98.65	**99.18**
	Cotton boll	20	85.03	**96.32**
	Cotton stem	52.36	83.33	**90.66**
	Overall	57.23	90.1	**95.22**
mIoU (%)	Cotton leaf	61.3	90.4	**95.11**
	Cotton boll	14	68.83	**84.11**
	Cotton stem	19.06	64.87	**80.37**
	Overall	26.79	74.69	**86.53**

The segmentation results were evaluated in terms of four dimensions: accuracy (higher is better), F1 score (higher is better), and mIoU

The bold numbers indicate the best results in the respective category

Qualitative comparison

The point cloud plants were segmented into leaves, bolls, and branches at the seedling, bud, and boll stages of the cotton growth cycle. The visualization results of point cloud organ segmentation of cotton plants show that TPointNetPlus outperforms PointNet++ and PointNet in segmenting main stems, branches, and bolls, as shown in Fig. 7. The segmentation results of PointNet validate the quantitative segmentation results in Table 4. PointNet performs poorly on all three metrics reviewed, and Fig. 7 shows that most points were misclassified. From leaf and stem segmentation at the seedling stage (Fig. 7a), to leaf, boll, and stem segmentation at the bud stage (Fig. 7b), and to leaf, boll, and stem segmentation at the boll stage (Fig. 7c), more misclassifications occur. Compared to the real values, PointNet++ and TPointNetPlus have some errors in organ segmentation, but the overall segmentation effect is good. PointNet++ has errors in branch and leaf segmentation, especially at the top of the plant at the boll stage. This is due to the thinness of the branches and trunks at the top of the cotton plant, and the insufficient extraction of network features.

The results of the cotton plant organ segmentation visualization showed that TPointNetPlus and PointNet++ both achieved overall accuracies greater than 90%. However, there were several false predictions in both networks. TPointNetPlus misclassified a part of the branch stem as a leaf in some cases (Fig. 8d), while the PointNet++ misprediction was more prominent (Fig. 8c). Additionally, parts of the branches were incorrectly predicted as leaves. These misclassified branch regions were connected to the leaves and had thin branches, but this misclassification was not as prominent in the TPointNetPlus inference as it was in the PointNet++ inference. Moreover, TPointNetPlus successfully segmented cotton bolls, while the other two models did not.

Another source of incorrect predictions is incorrect manual labelling. Due to manual labelling, many leaf stalks attached to leaves are labelled leaves, and the network learns to segment them as leaves. As a result, in some cases, the branch portion attached to the leaf blade was incorrectly segmented. Similarly, the network incorrectly categorizes thicker branches as leaves. This is because they are similar in shape to smaller blade sections. However, due to their small size, they were manually labelled as part of the branch in the ground truth.

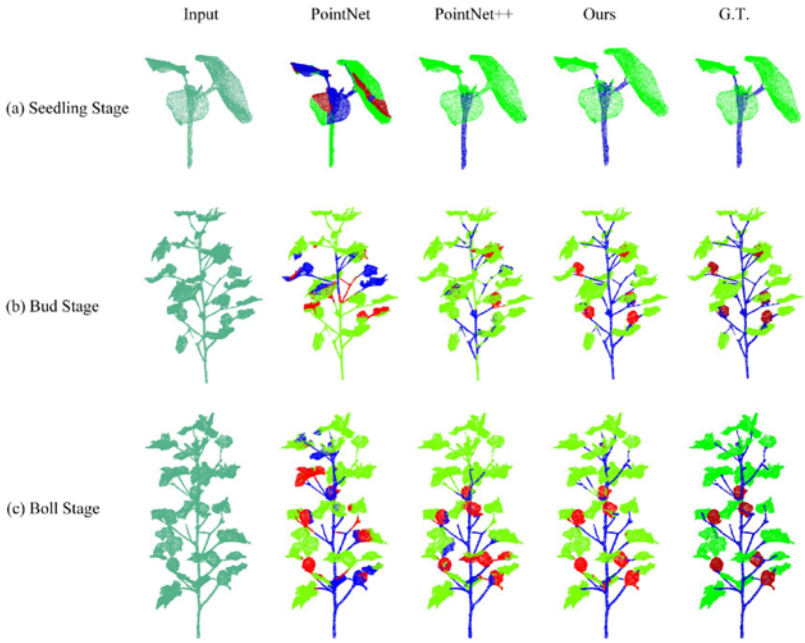


Fig. 7 Comparison of segmented point clouds from the three deep learning models with the ground truth. Predicted segments from TPointNetPlus, PointNet++, and PointNet for cotton bolls, leaves, and stems are shown in red, green, and blue, respectively

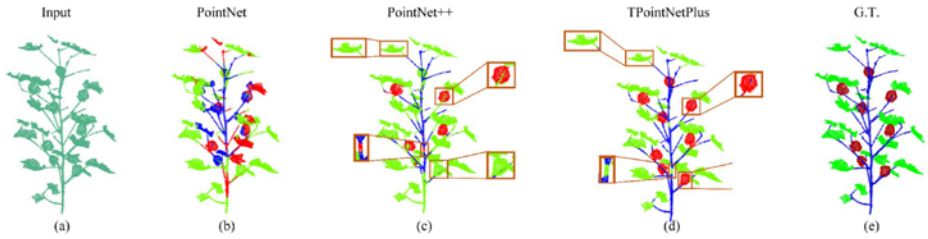


Fig. 8 Comparative visualization of the segmentation performance of three deep learning networks on leaves, bolls, and stems. The boxes show the segmentation details of the different methods on the different organs

Results of instance segmentation

The TPointNetPlus network was used to perform semantic segmentation on a cotton plant point cloud, resulting in data on leaves, cotton bolls, and branches. To further distinguish individual leaves and cotton bolls, the HDBSCAN algorithm was used to conduct instance segmentation. Figure 9 illustrates the successful segmentation of cotton plants in the boll stage, these plants had more than 15 leaves and 5 or more cotton bolls. The successful semantic and instance segmentation of leaves and cotton bolls provides an ideal basis for accurate measurements in the following stages.

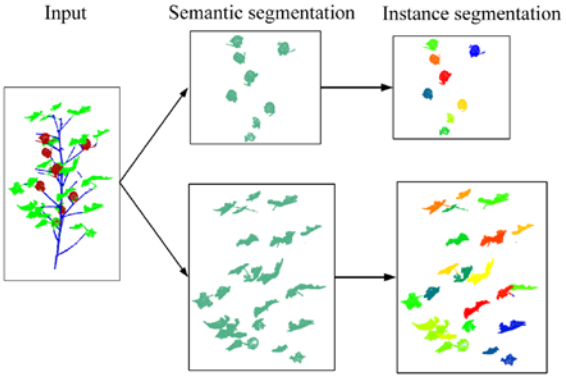


Fig. 9 Cotton plant instance segmentation procedure. Different cotton bolls and leaves are displayed in different colours, which in turn completes the instance segmentation task

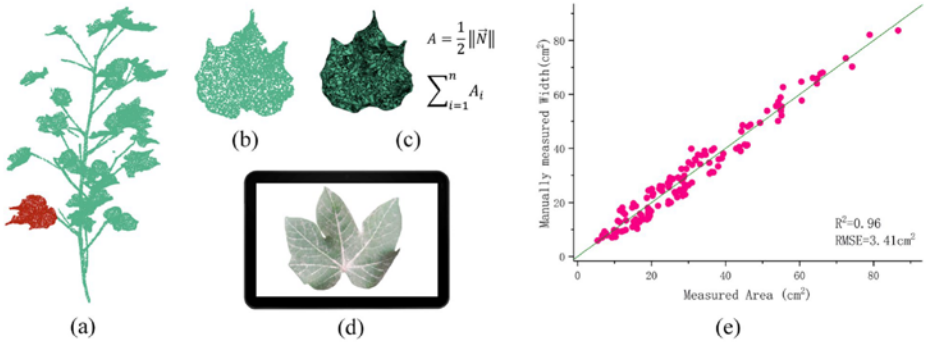


Fig. 10 Cotton leaf area trait extraction and correlation with the ground truth. **a** Postprocessed sample from the test set. A cotton leaf is segmented in red. **b** A complete point cloud dataset of a leaf. **c** The point cloud of the leaf is triangulated, the area of each triangle mesh is calculated, and the areas of all the triangles are summed to obtain the total surface area of the entire surface. **d** Obtaining the ground truth values using the LA-S series plant imager by Wanshen. **e** Correlation of leaf area extracted from predicted and ground truth segments

Results of phenotypic parameter extraction

Cotton leaf area

Figure 10 illustrates the process and analysis of measuring the cotton leaf area. Initially, a cotton leaf was segmented from the cotton plant point cloud (Fig. 10a). The actual leaf area was measured using an LA-S series plant image analyser (Fig. 10d). The predicted value was obtained by triangulating the point cloud of the leaf (Fig. 10b) to form a triangular mesh representation of the leaf (Fig. 10c). Across the entire test dataset, the estimated leaf area based on the predicted segment showed a high correlation with the actual measurements, with an R^2 value exceeding 0.96 (Fig. 10e). For leaf area, the root mean square error (RMSE) was relatively low, with an RMSE = 3.41, indicating accurate predictions of leaf area size in most cases. A comparison between the ground truth values and predicted values demonstrated that the estimated leaf area was generally equal to or smaller than the ground truth values, without exceeding the actual measured values (Fig. 10e). Since the estimation of leaf area relies on the triangular mesh, accurate pediction of these points is crucial, and occasional rare segmentation errors in the leaf area do not affect the overall area calculation.

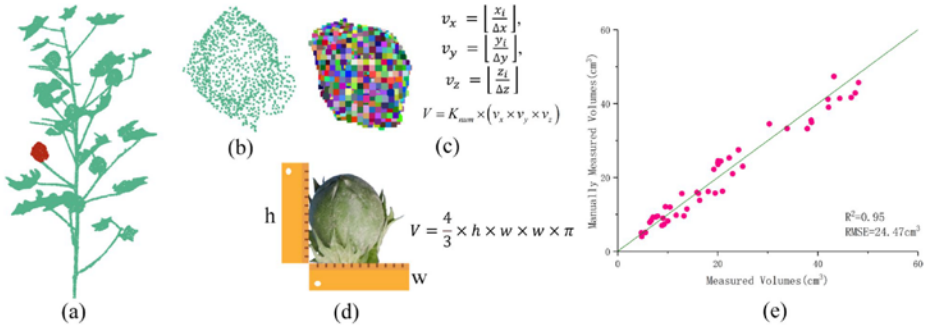


Fig. 11 Cotton boll trait extraction and correlation with the ground truth. **a** Postprocessed sample from the test set. The cotton boll is segmented in red. **b** A complete point cloud dataset of a boll. **c** The point cloud is voxelated, representing each voxel's volume by the contained points. The cotton boll total volume was computed by summing the individual voxel volumes. **d** The boll volume of the cotton plants was manually measured using a ruler. **e** Correlation of cotton boll volume extracted from predicted and ground truth segments

Boll volumes

The process of measuring the volume of cotton bolls is demonstrated in Fig. 11. The test data showed that accurate predictions were made (Fig. 11a). The volumetric representation of the cotton boll point cloud, obtained through voxelization, enables the calculation of the boll volume based on the point cloud density (Fig. 11b). The leaf area traits derived from the predicted segments were strongly correlated with the ground truth measurements, with an R^2 value greater than 0.95 (Fig. 11e). The root mean square error (RMSE) for boll volume was low at RMSE = 24.47, indicating precise predictions of cotton boll volume in most cases. A comparison between the ground truth values and predictions showed that the estimated boll volume was usually equal to or lower than the ground truth values without surpassing the actual measurements (Fig. 11e).

Leaf number

The process of counting cotton leaves and analyzing the results is illustrated in Fig. 12. Initially, the TPointNetPlus network segments the point cloud model of the cotton plant to identify leaves (Fig. 12a). Subsequently, each leaf is extracted using the HDBSCAN clustering algorithm (Fig. 12b). Finally, the number of leaves is statistically analyzed using the counting function (Fig. 12c). The estimated number of leaves based on the predicted segments shows a high correlation with the actual values, with an R-squared value exceeding 0.98 (Fig. 12d). The root mean square error (RMSE) for leaf number is relatively low at RMSE = 0.62, indicating accurate prediction of leaf number in most cases. The experimental results demonstrate that this process can effectively and accurately identify and quantify the leaves of cotton plants, providing valuable data for further research and analysis.

Discussion

3D reconstruction of cotton plants

3D reconstruction of cotton plants is essential for acquiring point cloud data, with its quality directly affecting data accuracy. Currently, different 3D reconstruction techniques have been used to construct 3D plant models for phenotyping [42]. Although LiDAR scanners can provide highly accurate point cloud data, they may be affected by plant density and shading [43]. Dense vegetation may cause the laser beam to lose some information as it passes through the plant layers, and occlusion between plants may complicate the challenge of obtaining a complete plant structure. A 3D time-of-flight camera (ToF) has been shown to rapidly acquire 3D images of plants, but its resolution and accuracy relatively low [44–46]. 3D laser scanning typically acquires data from only one point of view and therefore may present challenges when dealing with complex plant structures [43, 47, 48]. This single viewpoint may not be able to capture all the details of the plant surface, especially if the plant's underside or leaves overlap. The contour shape-based method is efficient in measuring plant volume, stem height, and surface area of individual leaves, but it may not be robust enough for localized occlusion or leaf overlap, leading to a decrease in the accuracy of the measurements [49].

The multi-view point based approach [50] examines plants from multiple perspectives simultaneously, providing more comprehensive and accurate information about their structure. This method addresses issues such as occlusion and lack of detail that can arise from a single

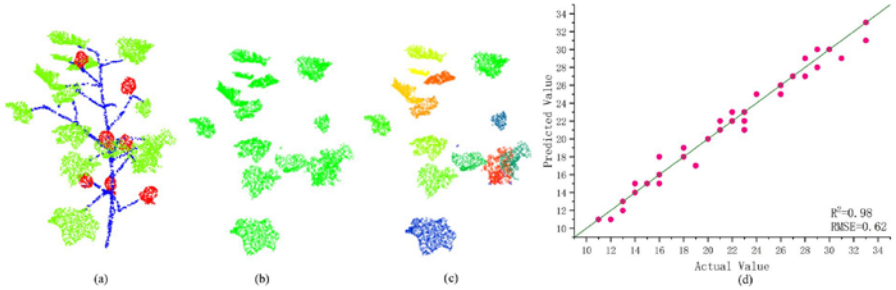


Fig. 12 Cotton leaf number statistics and correlation with the ground truth. **a** Cotton plant 3D point cloud model. **b** Leaf point cloud segmentation results. **c** Instance segmentation results for leaf. **d** Correlation of Leaf number statistics from predicted and ground truth segments

viewpoint. However, the point cloud collection method used in this article has limitations: it cannot be applied on a large scale outdoors and requires destructive sampling. Additionally, collecting 180 images involves a significant workload and the reconstruction operation time is lengthy. Despite these drawbacks, the cotton multi-view plant 3D reconstruction method achieves more precise depth information and integrates geometric morphology with texture information to create a more realistic and lifelike 3D model of the cotton plant. This method can assist botanists and ecologists in better studying the growth patterns and interactions of cotton, deepening their understanding of its structure and function, and diagnosing pests and diseases.

Cotton plant height

In this study, we measured and analyzed the height of cotton plants. However, the dataset only includes 5 sets of data, which is relatively small and may affect the reliability and representativeness of the results. The following will analyze the reasons for the insufficient data and its impact on the study's findings, as well as propose potential improvements for future research.

The process of measuring plant height in the tested cotton plants and analysing the results are illustrated in Fig. 13. The prediction accuracy was consistent across all test cases, accurately predicting the bottom 1 cm region of the main stem (Fig. 13b). There was a high correlation between the main stem traits estimated from the predicted segments and those estimated from the ground truth throughout the entire test set, with R^2 values exceeding 0.97 (Fig. 13d). RMSE was low (RMSE = 0.26), indicating accurate categorization of the lowest and highest points in most cases. A comparison between the ground truth and predicted values demonstrated that the estimated plant height values were equal to or less than the ground truth values and did not exceed them

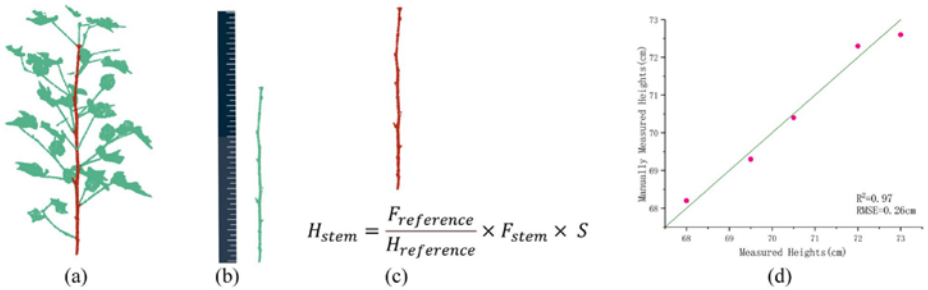


Fig. 13 Main stem trait extraction and correlation with the ground truth. **a** Postprocessed sample from the test set. The main stem is segmented in red. **b** The height of the cotton plants was manually measured using a ruler. **c** Mainstem height estimation and selection of the bottom 1 cm region. **d** Correlations of main stem diameter and height extracted from the predicted and ground truth segments

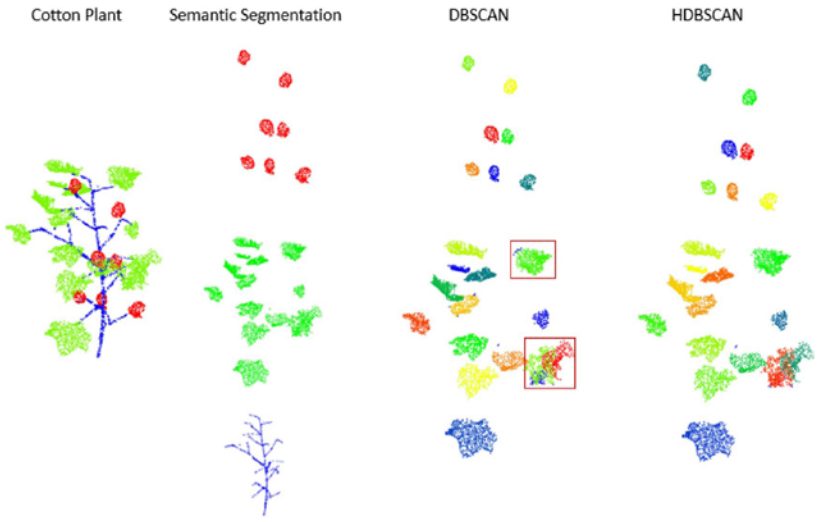


Fig. 14 The clustering algorithms DBSCAN and HDBSCAN instance segmentation results comparison

(Fig. 13c). Since plant height estimation relies on correctly predicting the highest and lowest points, any occasional missegregations in the middle of the plant height had no impact on the resulting heights.

Analysis of the clustering results

Density-based spatial clustering of applications with noise (DBSCAN) as a density clustering algorithm has shown significant advantages in many aspects. Its robustness enables it to handle noise and outliers efficiently without the need to know the number of clusters in advance [51]. Compared to other clustering algorithms, DBSCAN has better adaptability to irregularly shaped clusters, is able to discover clusters of arbitrary shapes, and is sensitive to data with large variations in density [52]. In addition, the algorithm performs well in terms of scalability and is suitable for clustering tasks on large-scale datasets [51]. However, DBSCAN has several drawbacks. Its performance is sensitive to clusters with large differences in density in the data, and the parameters may need to be adjusted to accommodate clusters with different densities [53]. In addition, DBSCAN may face the challenge of dimensionality catastrophe when dealing with high-dimensional data and needs to be used with caution [54].

Compared to HDBSCAN, the main disadvantage of DBSCAN is the need to prespecify some parameters, such as the neighborhood radius and the number of points in the minimum neighborhood, whereas HDBSCAN is relatively more adaptive and does not require an explicit density threshold [55]. HDBSCAN is also hierarchical in nature and is able to perform clustering at different density levels, making it more suitable for clusters with different density levels [56]. Both DBSCAN and HDBSCAN yield correct results when the cotton instance is split. However, DBSCAN was incorrect when segmenting leaf instances with more complex morphology and closer proximity (Fig. 14).

Conclusion

The proposed TPointNetPlus, a cotton point cloud organ instance segmentation method, seamlessly integrates deep learning and clustering algorithms to enhance the accuracy of phenotypic organ structure measurements through 3D point clouds. The creation of a dedicated point cloud dataset for cotton plants, coupled with the incorporation of the attention module transformer into the PointNet++model, contributes to precise feature extraction. The application of the HDBSCAN algorithm for organ-level cotton plant point cloud segmentation successfully isolates cotton leaves, bolls, and branches, providing accurate phenotypic feature parameters. The research outcomes highlight the exceptional semantic segmentation accuracy of TPointNetPlus (98.38%) for cotton leaves. Correlation coefficients between the measured values of key phenotypic parameters demonstrated

the model's reliability in predicting traits such as plant height, leaf area, and boll volume. This automated method involves translating a plant's 3D point cloud into phenotypic parameters, which can be applied in fields such as cotton breeding and plant physiology.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13007-025-01357-w.

Supplementary Material 1.

Author contributions

CJS and FYL wrote the manuscript with contributions from all authors. SQS performed the experiments. CJS analyzed the results. HG collected data on cottons. LYS set up a simple experimental setup. HG performed the image analysis. All authors have revised and approved the final manuscript.

Funding

This work was supported by the Ministry of Education Industry-University Cooperation Collaborative Education Program (No. 220505876092353), the First-class Undergraduate Programmes Foundation in Computer Graphics at Tarim University (No. TDYLKC202231), and the Xinjiang Production and Construction Corps Science and Technology Program (No. 2022DB002).

Availability of data and materials

No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate

This research contains no materials, procedures, and case studies related to human and/or animal.

Competing interests

The authors declare no competing interests.

Received: 13 January 2024 Accepted: 4 March 2025
Published online: 16 March 2025

References

1. Li N, et al. Impact of climate change on cotton growth and yields in Xinjiang, China. Field Crops Res. 2020. https://doi.org/10.1016/j.fcr.2019.107590.
2. Lin HX, et al. Rationality of Xinjiang to undertake textile industry based on water resources carrying capacity. Desalin Water Treat. 2021;219:147–56.
3. Pabuayon ILB, et al. High-throughput phenotyping in cotton: a review. J Cotton Res. 2019. https://doi.org/10.1186/s42397-019-0035-0.
4. Das Choudhury S, et al. Leveraging image analysis to compute 3D plant phenotypes based on voxel-grid plant reconstruction. Front Plant Sci. 2020. https://doi.org/10.3389/fpls.2020.521431.
5. Yang WN, et al. Crop phenomics and high-throughput phenotyping: past decades, current challenges, and future perspectives. Mol Plant. 2020;13(2):187–214.
6. Li ZB, et al. A review of computer vision technologies for plant phenotyping. Comput Electron Agric. 2020. https://doi.org/10.1016/j.compag.2020.105672.
7. Khan Z, et al. Estimation of vegetation indices for high-throughput phenotyping of wheat using aerial imaging. Plant Methods. 2018. https://doi.org/10.1186/s13007-018-0287-6.
8. Huang LW, et al. Real-time motion tracking for indoor moving sphere objects with a LiDAR sensor. Sensors. 2017. https://doi.org/10.3390/s17091932.
9. Perez AJ, Perez-Cortes JC, Guardiola JL. Simple and precise multi-view camera calibration for 3D reconstruction. Comput Ind. 2020. https://doi.org/10.1016/j.compind.2020.103256.
10. Zhong YJ, et al. Multi-view 3d reconstruction from video with transformer. In IEEE International Conference on Image Processing (ICIP). 2022. Bordeaux, France.
11. Xie CQ, Yang C. A review on plant high-throughput phenotyping traits using UAV-based sensors. Comput Electron Agric. 2020. https://doi.org/10.1016/j.compag.2020.105731.
12. Feng L, et al. A comprehensive review on recent applications of unmanned aerial vehicle remote sensing with various sensors for high-throughput plant phenotyping. Comput Electron Agric. 2021. https://doi.org/10.1016/j.compag.2021.106033.
13. Qi CR, et al. PointNet: deep learning on point sets for 3D classification and segmentation. In 30th IEEE/CVF conference on computer vision and pattern recognition (CVPR). 2017. Honolulu, HI.
14. Thomas H, et al. KPConv: flexible and deformable convolution for point clouds. In IEEE/CVF international conference on computer vision (ICCV). 2019. Seoul, South Korea.
15. Wang F, Bryson M. Tree segmentation and parameter measurement from point clouds using deep and handcrafted features. Remote Sens. 2023. https://doi.org/10.3390/rs15041086.
16. Sun P, Yuan X, Li D. Classification of individual tree species using UAV LiDAR based on transformer. Forests. 2023. https://doi.org/10.3390/f14030484.
17. Saeed F, et al. Cotton plant part 3D segmentation and architectural trait extraction using point voxel convolutional neural networks. Plant Methods. 2023;19(1):33.
18. Ruiming D, et al. PST: plant segmentation transformer for 3D point clouds of rapeseed plants at the podding stage. J Photogramm Remote Sens. 2023;195:380–92.
19. Li H, et al. Automatic branch-leaf segmentation and leaf phenotypic parameter estimation of pear trees based on three-dimensional point clouds. Sensors. 2023. https://doi.org/10.3390/s23094572.
20. Liu B, et al. TSCMDL: multimodal deep learning framework for classifying tree species using fusion of 2-D and 3-D features. IEEE Trans Geosci Remote Sens. 2023;61:1–11.
21. Meng HY, et al. VV-NET: Voxel VAE net with group convolutions for point cloud segmentation. In IEEE/CVF International Conference on Computer Vision (ICCV). 2019. Seoul, South Korea.
22. Xu MT, et al. PAConv: position adaptive convolution with dynamic kernel assembling on point clouds. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021. Electr Network.
23. Wang Y, et al. Dynamic graph CNN for learning on point clouds. ACM Trans Graph. 2019;38(5):1–12.
24. Zeng Z, et al. RG-GCN: a random graph based on graph convolution network for point cloud semantic segmentation. Remote Sens. 2022. https://doi.org/10.3390/rs14164055.
25. Xu QG, et al. Grid-GCN for fast and scalable point cloud learning. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020. Electr Network.
26. Wang F, et al. Residual attention network for image classification. In 30th IEEE/CVF conference on computer vision and pattern recognition (CVPR). 2017. Honolulu, HI.
27. Wu B, et al. Visual transformers: token-based image representation and processing for computer vision. 2020. abs/2006.03677.
28. Dosovitskiy A, et al. An image is worth 16x16 words: transformers for image recognition at scale. 2020. abs/2010.11929.
29. Zhang H, et al. Self-attention generative adversarial networks. in 36th international conference on machine learning (ICML). 2019. Long Beach, CA.
30. Carion N, et al. End-to-end object detection with transformers. in European conference on computer vision. 2020. Springer.
31. Wan J, et al. DGANet: a dilated graph attention-based network for local feature extraction on 3D point clouds. Remote Sens. 2021. https://doi.org/10.3390/rs13173484.
32. Feng MT, et al. Point attention network for semantic segmentation of 3D point clouds. Pattern Recogn. 2020. https://doi.org/10.1016/j.patcog.2020.107446.

33. Wen X, et al. CF-SIS: semantic-instance segmentation of 3D point clouds by context fusion with self-attention. In Proceedings of the 28th ACM International Conference on Multimedia. 2020. p. 1661–1669.
34. Niu Z, Zhong G, Yu H. A review on the attention mechanism of deep learning. Neurocomputing. 2021;452:48–62.
35. Guo M-H, et al. Attention mechanisms in computer vision: a survey. Comput Visual Media. 2022;8(3):331–68.
36. Wen X, et al. PMP-Net++: point cloud completion by transformer-enhanced multi-step point moving paths. IEEE Trans Pattern Anal Mach Intell. 2023;45(1):852–67.
37. Dosovitskiy A, et al. An image is worth 16x16 words: transformers for image recognition at scale. 2020.
38. Qi CR, et al. PointNet plus plus: deep hierarchical feature learning on point sets in a metric space. In 31st Annual Conference on Neural Information Processing Systems (NIPS). 2017. Long Beach, CA.
39. Campello R, et al. Hierarchical density estimates for data clustering, visualization, and outlier detection. ACM Trans Knowl Discov Data. 2015. https://doi.org/10.1145/2733381.
40. Ho Y, Wookey S. The real-world-weight cross-entropy loss function: modeling the costs of mislabeling. IEEE Access. 2020;8:4806–13.
41. Zhou DF, et al. IoU Loss for 2D/3D Object Detection. In 7th International Conference on 3D Vision (3DV). 2019. Quebec City, Canada.
42. Shi WN, et al. Plant-part segmentation using deep learning and multi-view vision. Biosyst Eng. 2019;187:81–95.
43. Thapa S, et al. A novel LiDAR-based instrument for high-throughput, 3D measurement of morphological traits in maize and sorghum. Sensors. 2018. https://doi.org/10.3390/s18041187.
44. Chaivivatrakul S, et al. Automatic morphological trait characterization for corn plants via 3D holographic reconstruction. Comput Electron Agric. 2014;109:109–23.
45. Guan HO, et al. Three-dimensional reconstruction of soybean canopies using multisource imaging for phenotyping analysis. Rem Sens. 2018. https://doi.org/10.3390/rs10081206.
46. Vázquez-Arellano M, et al. Leaf area estimation of reconstructed maize plants using a time-of-flight camera based on different scan directions. Robotics. 2018. https://doi.org/10.3390/robotics7040063.
47. Paulus S, et al. Low-cost 3d systems: suitable tools for plant phenotyping. Sensors. 2014;14(2):3001–18.
48. Garrido M, et al. 3D maize plant reconstruction based on georeferenced overlapping LiDAR point clouds. Rem Sens. 2015;7(12):17077–96.
49. Golbach F, et al. Validation of plant part measurements using a 3D reconstruction method suitable for high-throughput seedling phenotyping. Mach Vis Appl. 2016;27(5):663–80.
50. Pound MP, et al. Automated recovery of three-dimensional models of plant shoots from multiple color images. Plant Physiol. 2014;166(4):1688-U801.
51. Ester M, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In kdd. 1996.
52. Ankerst M, et al. OPTICS: Ordering points to identify the clustering structure. In 1999 ACM SIGMOD International Conference on Management of Data. 1999. Philadelphia, Pa.
53. Schubert E, et al. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. ACM Trans Database Syst. 2017. https://doi.org/10.1145/3068335.
54. Kriegel HP, et al. Outlier detection in axis-parallel subspaces of high dimensional data. In 13th Pacific-Asia Conference on Knowledge and Data Mining. 2009. Bangkok, Thailand.
55. Campello RJ, Moulavi D, Sander J. Density-based clustering based on hierarchical density estimates. In Pacific-Asia conference on knowledge discovery and data mining. 2013. Springer.
56. McInnes L, Healy J. Accelerated hierarchical density based clustering. In 17th IEEE International Conference on Data Mining (ICDMW). 2017. New Orleans, LA.

Publisher's Note

https://doi.org/10.1186/s13007-025-01357-w

4. 软件著作权：席亚军. 图书馆座位智能管理 APP V1.0. 国家知识产权局, 2024.

中华人民共和国国家版权局
计算机软件著作权登记证书

证书号：软著登字第13281612号

软件名称：图书馆座位智能管理APP V1.0

著作权人：塔里木大学;席亚军

权利取得方式：原始取得

权利范围：全部权利

登记号：2024SR0877739

根据《计算机软件保护条例》和《计算机软件著作权登记办法》的规定，经中国版权保护中心审核，对以上事项予以登记。

中华人民共和国国家版权局 计算机软件著作权登记专用章

2024年06月26日