

# 移动互联网信息无障碍研究综述

刘华斌, 于金艳, 宋申苧, 张梦玺

(吉林大学 计算机科学与技术学院, 长春 130012)

**摘要:** 移动互联网信息无障碍(mobile Internet information accessibility, MIIA)旨在确保移动应用内容对所有用户(包括视障人士等)都能平等、便捷、无障碍地获取和使用。系统综述移动互联网信息无障碍领域的最新研究进展,重点分析总结移动端 GUI(graphical user interface)语义表征与理解、无障碍检测以及布局修复等方面的研究成果。分析表明,从传统启发式规则方法到深度学习驱动的自动化工具,相关技术逐渐提升了检测的精度和适应性,同时也揭示了在应对复杂动态交互和多样化用户需求方面的挑战,并对未来研究方向进行了展望。移动互联网信息无障碍技术已显著改善了视障用户的数字体验,但仍需不断创新与优化,以实现真正普惠与包容的数字社会。

**关键词:** 移动互联网; 信息无障碍; 视障用户; 图形用户界面

**中图分类号:** TP311 **文献标志码:** A **文章编号:** 1671-5489(2025)01-0124-15

## Review of Mobile Internet Information Accessibility Research

LIU Huaxiao, YU Jinyan, SONG Shenning, ZHANG Mengxi

(College of Computer Science and Technology, Jilin University, Changchun 130012, China)

**Abstract:** The purpose of mobile Internet information accessibility (MIIA) was to ensure that mobile application content was equally accessible, convenient, and barrier-free for all users, including those with visual impairments. We systematically review the latest research progress in the field of mobile Internet information accessibility, focusing on the analysis and summary of research achievements in semantic representation and understanding of mobile GUI, accessibility detection and layout repair. The analysis shows that from traditional heuristic rule methods to deep learning-driven automated tools, related technologies have gradually improved detection accuracy and adaptability, while also revealing challenges in addressing complex dynamic interactions and diverse user needs. We have provided an outlook on future research directions. MIIA technologies have significantly improved the digital experience for visually impaired users, but they still need continuous innovation and optimization to achieve a truly inclusive digital society.

**Keywords:** mobile Internet; information accessibility; visually impaired users; graphical user interface

## 0 引言

移动互联网信息无障碍(mobile Internet information accessibility, MIIA)主要指移动应用(App)

收稿日期: 2024-12-02.

**第一作者简介:** 刘华斌(1986—),男,汉族,博士,副教授,博士生导师,从事智能化软件工程、数据挖掘和人机交互的研究, E-mail: liuhuaxiao@jlu.edu.cn. **通信作者简介:** 宋申苧(2000—),男,汉族,硕士研究生,从事软件工程的研究, E-mail: sunning2118@gmail.com.

**基金项目:** 吉林省自然科学基金(批准号: 20230101070JC).

内容对任何人(无论是健全人还是残疾人、老年人)在任何情况下都能平等、方便、无障碍地获取、交互和使用<sup>[1-4]</sup>。目前,移动互联网逐渐成为公民获取政治、教育、体育等信息的主要来源,这对移动互联网信息的可访问性提出了更高的要求。因此,在移动应用发展过程中,不仅要迎合市场需求,还应兼顾不同人群信息获取能力的差异,致力于为所有用户提供一个平等、开放的数字空间,促使信息的共享和获取更普惠。

视障人群指患有视力障碍,包括完全失明或部分失明的群体。该群体可能依赖非视觉方法(如盲杖、导盲犬、声音提示等)或辅助技术弥补视觉上的缺失,以更好地适应并参与日常生活、工作和社会活动。视障用户在使用互联网获取信息时通常需要特殊设计和功能,以确保他们有与健全用户相当的体验和获取能力,从而需要在移动互联网开发中采取措施,使应用内容对视障用户更友好且易于访问。1999年,万维网联盟(world wide Web consortium, W3C)发起了网络无障碍计划(Web accessibility initiative, WAI)<sup>[5]</sup>,以改善网络的可访问性。WAI倡议负责制定一套重要的可访问性指南,称为网络内容可访问性指南(Web content accessibility guideline, WCAG)<sup>[6]</sup>。移动互联网信息无障碍为视障用户带来了巨大便利,使他们能更自主地获取信息、参与社交和独立生活。通过语音辅助技术,他们可以访问新闻、电子书等资源,丰富知识并促进个人发展;使用社交媒体和通讯应用,参与线上沟通;借助无障碍设计的导航应用,更安全地出行。移动互联网信息无障碍技术的进步显著提升了视障用户的生活质量和幸福感。根据《柳叶刀》最新调查显示<sup>[7]</sup>,中国是世界上视障人群占比最高的国家,视障群体约有5 928万人。此外,中国60岁及以上老年人超过2.9亿,社会对无障碍环境的需求广泛而迫切。如何为视障人群打开新“视”界,成为目前消除“数字鸿沟”,推进互联网信息无障碍化建设<sup>[8-9]</sup>,实现我国“十四五”规划愿景的重点工作之一<sup>[10-11]</sup>。

但互联网开发人员在设计和实现移动应用时常会忽略产品的无障碍需求,只为追求移动应用美观的视觉效果,缺乏对视障用户使用体验的关注。这种单一视角的设计理念可能导致视障用户无法实现信息的顺利获取,大幅度降低他们与互联网的交互流畅度,限制视障用户在数字环境中的平等参与。Bi等<sup>[12]</sup>于2021年从业者的角度和不同的软件开发阶段进行分析,发现可访问性并没有正确地集成到通用软件项目中。同时大多数的开发者没有相关的技能和知识,团队资源、公司能支持的专业知识和时间预算等也影响了实践中的可访问性设计与开发。根据在线调查<sup>[13]</sup>和实地半结构化访谈<sup>[14]</sup>了解到,仅有10%的受访者对当前App表示满意,而超过80%的受访者在使用App时会遇到诸多不便,其中尤以图形用户界面(GUI)组件外观不合理、组件导航顺序混乱、反馈提示不一致,以及移动应用中的GUI关键操作路径逻辑不自洽最突出,这些现象导致GUI组件难以准确识别和操作、视障用户易在应用中迷失方向、难以建立稳定的操作模式等问题,严重影响了视障用户对移动应用的整体可用性和友好性,给视障用户在使用移动应用时带来了困扰和不确定感。Fok等<sup>[15]</sup>于2022年分析了312个安卓应用程序在16个月中存在的缺失标签的可访问性问题,发现应用程序存在的无障碍问题不会随下载次数增加而改善,也并不与开发公司的规模相关。

移动互联网信息无障碍这一主题在移动互联网开发过程中的关注和优化能帮助视障人群进一步平等、方便地使用互联网。例如,盲人用户在使用移动端访问互联网内容时常会借助于屏幕阅读软件,而屏幕阅读软件仅关注页面中的文本信息而无法将页面中的布局、颜色等其他信息呈现给盲人用户。因此,一些布局欠佳的页面常会阻碍盲人用户对页面关键内容的理解。在移动互联网信息无障碍的实践过程中,可通过对页面进行检测提出合理的检测结果,以供互联网开发人员调整界面设计,使页面对盲人用户更友好。进一步,对于屏幕阅读软件无法呈现的图片,可对图片中的内容生成替代文本(Alt-Text),使盲人用户了解除页面中文本外更丰富的信息。但互联网信息无障碍的研究也面临诸多严峻挑战:首先,互联网信息的形式多种多样,包括文字、图片、音频、视频、图表等,每种形式对无障碍的设计要求均有不同,要提供完备的无障碍支持,显著增加了设计方案的复杂性;其次,在对无障碍设计进行评估时需从视障人群的真实需求出发进行评估,难以有统一的评估标准且因人而异;最后,互联网技术和用户界面设计快速发展,涌现出新的信息展现形式和交互模式,例如虚拟现实、增强现实等,互联网无障碍无法快速涵盖新技术并保持同步。

互联网无障碍的研究工作最早可追溯到互联网成立之初,通过 W3C 和 WAI 推动了无障碍标准的确立,为全球的互联网无障碍研究奠定了基础.目前,如何改善视障用户使用互联网的可访问性体验已成为国内外工业界和学术界的热点.近年来,腾讯、阿里、谷歌、苹果、字节跳动等大型互联网公司不断对产品进行无障碍优化,以增强产品面向视障用户的可访问性.目前,互联网无障碍研究已取得许多成果<sup>[16-28]</sup>.Chen 等<sup>[27]</sup>借助无障碍测试框架,构建了名为 XBot 的 GUI 可访问性检测工具,通过设计大量判定规则实现对可访问性问题更全面的覆盖.Zhang 等<sup>[28]</sup>于 2024 年针对 GUI(graphical user interface)布局中的低视力可访问性问题,包括组件尺寸小、间隔窄以及颜色对比度低,借助关系图卷积神经网络模型设计并实现了名为 AccessFixer 的工具,能准确且有效地为这些可访问性问题提供修复方案.在网页信息无障碍方面,李玉聪等<sup>[29]</sup>于 2024 年针对多样化网页上视障用户导航的难题,通过设计启发式规则采用决策树二分类算法,提出了一种自动标识导航栏地标的方法,显著提升了网站的无障碍性能,并改善了视障用户的使用体验.Gleason 等<sup>[30]</sup>提出了一个浏览器扩展工具 Twitter Ally,可通过 6 种方法在 Twitter 网站上为用户发布的图像提供高质量的自动描述,优化了视障用户对社交媒体平台的无障碍访问.为对互联网信息无障碍目前已有的研究工作进行系统分析总结,本文确定了 Internet Information Accessibility, Web Accessibility, Blind Users 等关键词,并基于 Web of Science 中的搜索引擎对已有研究结果按年份进行统计,结果如图 1 所示.由图 1 可见,从 2019 年开始,针对移动互联网信息无障碍的研究逐渐受到关注,仅 2023 年,相关的研究论文数量便突破 250 篇.尽管现有研究从多角度探讨了移动互联网的无障碍化问题,但尚缺乏系统性的综述对这些研究成果进行全面的总结和分析.因此,本文系统地分析总结了移动互联网信息无障碍的研究进展和实践成果,并分析了现有移动应用和服务在无障碍设计方面的优势与不足,以有助于研究人员进一步提升视障用户的使用体验.

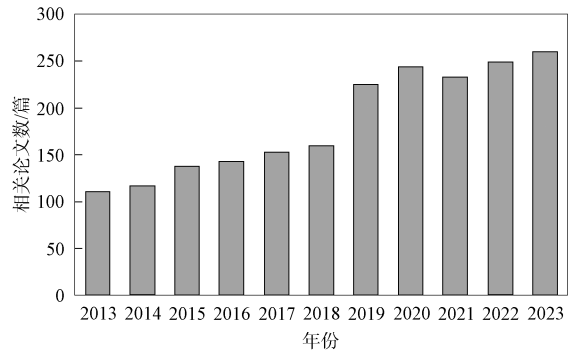


图 1 相关研究结果每年累计发表的论文数

Fig. 1 Cumulative number of published papers based on relevant research results for each year

## 1 移动端无障碍研究

随着移动设备和移动应用的普及,智能手机和平板电脑已成为人们日常生活中不可或缺的工具.根据国际电信联盟的统计,截至 2020 年,全球移动电话用户已超过 78 亿,几乎覆盖了全球人口<sup>[31]</sup>.但对有视觉障碍的用户以及老年人群体,移动设备的使用仍存在诸多障碍.因此,移动端无障碍成为确保视障人群和老年用户平等访问信息和服务的关键因素,也成为人机交互和软件工程领域的重要研究方向.移动设备因其特性(如触摸交互、屏幕尺寸限制和多种传感器)在无障碍设计上面临独特的挑战,移动端无障碍设计的目标是减少或消除视觉以及认知障碍用户在使用移动设备时所面临的困难.

### 1.1 移动端 GUI 的语义表征与理解

移动端 GUI 的语义表征与理解是 GUI 可访问性工作的基础.例如,视觉障碍用户在使用屏幕阅读器时,依赖于界面元素的语义标签理解各控件的功能.若无准确的语义信息,屏幕阅读器只能简单朗读“按钮”或“图片”,而无法传达具体的功能或内容,导致用户操作体验受阻.因此,通过语义表征界面元素能清晰地描述其功能和交互逻辑,为无障碍用户提供准确的使用信息.但移动端 GUI 的结构、功能模块和交互模式日趋丰富和复杂化,为移动端 GUI 的语义表征与理解提出了严峻挑战.例如,应用中嵌入了更多动态内容、多媒体元素以及复杂的交互逻辑,这些都增加了 GUI 语义理解的难度.

早期传统的启发式规则方法在移动端 GUI 的语义表征与理解中占有重要地位.基于启发式规则的方法依赖于预定义的经验规则集分析和解读移动端 GUI 中的元素特征.这些规则通常基于开发者

和设计人员的实践经验, 包含了一系列面向不同控件的可访问性指导. 例如, 一个启发式规则可能会规定: 所有具有“Button”标签的控件在无障碍工具中应被识别为可点击的按钮; 具有特定颜色或对比度的文本元素需要具备特定的颜色标准, 以确保视觉可读性, 使界面在一定程度上能满足无障碍需求. 但基于规则的方法存在扩展性差和适应性不足的问题. 当面对设计风格多样、元素复杂的界面时, 规则的编制和维护成本较高. 此外, 规则通常依赖于特定的领域或应用, 难以通用化. 随着深度学习和图像识别技术的发展, 基于数据驱动的深度学习的模型逐渐成为替代启发式规则的方法. 深度学习能自动学习控件的视觉特征和语义信息, 避免了启发式规则对预定义规则的依赖, 并且在动态内容和自定义控件识别上有显著优势.

在语义表征方面, 本文团队结合 GUI 截图和对应的布局结构文件, 提出一种将 GUI 转换为 GUI 拓扑图结构的方法, 有效表征了 GUI 内各组件的属性特征、层次关系和位置关系<sup>[28,32]</sup>. Chen 等<sup>[33]</sup>提出了一个深度学习模型 LabelDroid, 通过图像识别和自然语言生成技术, 自动为移动应用中基于图像的按钮生成内容描述, 这是首个专门用于解决 GUI 元素标签缺失问题的自动化模型. LabelDroid 模型采用了卷积神经网络(CNN)与 Transformer 模型相结合的编码-解码架构, 用于自动提取图像特征并生成自然语言标签. 这种架构不仅适合图像特征提取, 还具备处理长序列文本生成的能力, 有助于提升标签生成的准确性和多样性. 此外, 研究人员构建了一个大规模的无障碍性数据集, 包含来自上万种应用的图像按钮及其高质量的内容描述. 实验结果表明, LabelDroid 模型在预测标签的准确性上优于传统的启发式方法和现有的基准模型, 可减少互联网开发人员在应用开发中需要理解和遵循无障碍指南的工作量. Mehralian 等<sup>[34]</sup>在分析了相同图标可能有不同语义的情况对 LabelDroid 模型的影响后提出了一种上下文感知的标签生成方法 COALA, 在生成准确标签时综合考虑了来自图标的多种信息来源. COALA 采用了预训练的卷积神经网络模型 ResNet 进行图像编码, 并使用热编码器和单词嵌入模型 GloVe 进行上下文编码, 然后使用长短期记忆网络(LSTM)进行解码, 使其能逐个迭代地生成标签. Li 等<sup>[35]</sup>提出了 Screen2Vec, 一种生成移动端 GUI 屏幕和组件语义嵌入的自监督方法. 通过训练模型预测屏幕和组件在用户交互轨迹中的位置关系, 生成 GUI 屏幕和组件的语义嵌入. Screen2Vec 借鉴了 Word2Vec<sup>[36]</sup>的思路, 利用连续词袋模型(CBOW)预测屏幕和组件上下文, 以捕捉 GUI 元素的语义关系. Screen2Vec 将文本内容、视觉布局和应用元数据(如应用商店描述)整合进模型的语义嵌入中. 这种多模态表征能捕捉更丰富的语义信息, 使模型在处理视觉相似但语义不同的屏幕时表现更优. 通过多种下游任务展示了 Screen2Vec 的实用性, 包括相似界面检索、嵌入组合和任务嵌入.

Zhang 等<sup>[37]</sup>提出了一种利用像素图像直接推断移动应用界面元素的无障碍元数据方法, 无需依赖应用开发者提供的无障碍描述. 该方法可以为无障碍工具(如屏幕阅读器)生成并补充缺失的元数据, 从而提高了现有应用的无障碍性. 其通过构建一个高效、内存友好的对象检测模型并结合启发式规则设备端实时检测 GUI 元素. 模型能在 iPhone 上运行, 仅需约 10 ms 的推理时间, 使得它可以在用户设备上直接执行而无需云端计算. Xie 等<sup>[38]</sup>提出了一种基于心理学启发的无监督方法, 该方法首次在 GUI 感知分组中系统应用了 Gestalt 心理学原则, 包括连接性、相似性、邻近性和连续性等原则. 这些原则解释了人类如何基于视觉线索(如相似性和接近性)将元素分组, 从而帮助系统自动化地识别出人类在视觉上具有相似感知控件集合, 如列表、卡片、菜单等. 这种心理学启发的分组方式不同于传统的启发式方法, 它不依赖于预定义规则, 而是通过无监督方式实现高层次的 GUI 元素分组. 此外, 研究人员分析了感知分组在 GUI 设计搜索、模块化 GUI 代码生成和 UI 自动化测试中的应用潜力. 例如, 在 GUI 设计搜索中, 该方法可帮助开发者基于 GUI 结构而非仅视觉相似性找到相似设计, 显著提升了设计搜索的精度. 此外, 通过模块化感知分组信息的生成, 该方法支持更少冗余、可重用性强的 GUI 代码生成, 有助于实现快速的界面迭代和优化.

自然语言处理(NLP)技术在 GUI 语义理解中也发挥着重要作用. 通过对界面中的文本内容进行分析, 可获取元素的功能描述、提示信息 and 用户输入等语义信息. Fu 等<sup>[39]</sup>提出了“像素词到屏幕句”(pixel-words to screen-sentence, PW2SS)框架, 创新性地将 GUI 理解中的基本单位定义为“像素词”(pixel-words), 并将它们聚合成“屏幕句子”(screen-sentence). 基于 BERT 结构的屏幕 Transformer,

用于建模像素词之间的关系,以实现屏幕的整体理解.这种方法在视觉基础上完成了 GUI 理解,不再依赖屏幕元数据,解决了传统方法中元数据噪声大、获取困难等问题.除针对移动端 GUI 页面进行分析外,研究人员还对反馈的 GUI 问题报告进行了分析.GUI 的视频错误报告是指用户或测试人员通过录制屏幕视频的方式,记录下在移动端 GUI 遇到的错误或异常情况.这种报告形式不同于传统的文字描述,提供了更直观的视觉信息,使开发人员可以清楚地看到错误是如何出现的、出现在哪些交互步骤之后,这种报告在移动端 GUI 开发中越来越常用.Yan 等<sup>[40]</sup>提出了一种自动化识别这些重复的视频错误报告方法 Janus,它针对开发者在面对大量视频错误报告时遇到的繁琐和重复性问题,通过自动化检测相似的视频以显著减少冗余工作.为实现对视频中 GUI 场景的精准理解,Janus 将视频中的视觉信息、文本信息以及视频帧的顺序结合在一起,从多种信息源进行全面分析,可帮助开发者快速找到已知问题的重复报告,提升了错误修复和管理效率.这种方法和技术能用于开发自动化无障碍检测工具,从而提高无障碍性问题的检测和分析效率.

大语言模型(large language models, LLMs)<sup>[41]</sup>是基于深度学习技术的自然语言处理模型,它通过在大规模文本数据上进行训练,具有强大的语言理解、生成和推理能力.近年来,大语言模型(如 OpenAI 的 GPT 系列、Meta 的 LLaMA 等)备受关注,并迅速扩展到多模态领域,结合图像、音频等数据能处理跨越文本和视觉的任务.这种多模态能力进一步推动了模型在各应用场景中的使用,包括用户界面无障碍设计.Nong 等<sup>[42]</sup>提出了一个专为移动应用 GUI 代理设计的多模态大型语言模型 MobileFlow.它采用混合视觉编码器动态调整输入分辨率,以捕捉 GUI 界面的细节信息.这种能力对无障碍设计非常关键,因为它可以准确识别和描述界面元素的位置、大小和相对关系,从而为视力受限的用户提供清晰的界面导航描述.通过自然语言和视觉信息的融合,MobileFlow 可以根据用户指令生成适当的交互行为,如点击、滑动或输入.此外,传统的 GUI 代理通常需要通过系统 API 调用获取界面布局信息,可能带来隐私风险.MobileFlow 通过纯视觉感知的方法,能在不调用系统 API 的情况下解析界面布局,从而既保护了用户隐私,又可以为无障碍功能提供可靠的界面解析能力.

表 1 列出了移动端 GUI 的语义表征与理解.

表 1 移动端 GUI 的语义表征与理解

Table 1 Semantic representation and understanding of mobile GUI

方法	提出时间	主要特征
LabelDroid <sup>[33]</sup>	2020 年	通过图像识别和自然语言生成技术,自动为移动应用中基于图像的按钮生成内容描述
COALA <sup>[34]</sup>	2021 年	一种上下文感知的标签生成方法,通过 ResNet 编码图像、GloVe 编码上下文,并使用 LSTM 解码生成标签
Screen2Vec <sup>[35]</sup>	2021 年	一种生成移动端 GUI 屏幕和组件语义嵌入的自监督方法,通过训练模型预测屏幕和组件在用户交互轨迹中的位置关系,从而生成 GUI 屏幕和组件的语义嵌入
Screen recognition <sup>[37]</sup>	2021 年	一种利用像素图像直接推断移动应用界面元素的无障碍元数据方法,无需依赖应用开发者提供的无障碍描述
基于心理学启发的无监督方法 <sup>[38]</sup>	2022 年	一种系统应用了 Gestalt 心理学原则的 GUI 感知方法,通过无监督方式实现高层次的 GUI 元素分组
PW2SS <sup>[39]</sup>	2024 年	一个像素词到屏幕句框架,创新地将 GUI 理解中的基本单位定义为“像素词”,并将它们聚合成“屏幕句子”
Janus <sup>[40]</sup>	2024 年	一种自动化识别重复的视频错误报告方法,能自动地检测视频错误报告中 GUI 场景以显著减少开发者的冗余工作
MobileFlow <sup>[42]</sup>	2024 年	一个专为移动应用 GUI 代理设计的多模态大型语言模型,它采用混合视觉编码器动态调整输入分辨率,以捕捉 GUI 界面的细节信息

由表 1 可见,现有的移动端 GUI 语义表征与理解方法各有优劣.基于启发式规则的方法简单直观,适用于一般性界面元素的识别,但在处理复杂、多样化的动态 GUI 时,扩展性和适应性较差.深度学习方法通过自动学习控件的视觉特征和语义信息,能更准确地理解界面元素,避免了对预定义规则的依赖,但也面临训练数据不足、模型复杂度高和泛化能力有限的挑战.心理学启发的方法利用人

类视觉感知原理, 提升了 GUI 元素分组和理解的准确性. 大语言模型(如 MobileFlow)的引入, 融合了视觉和语言信息, 增强了对界面元素的语义理解. 但这些方法在模型复杂度、计算资源需求和实际部署方面仍存在困难. 未来, 增强移动端 GUI 的语义表征与理解, 可以通过开发高效的模型结构, 提高模型的泛化能力和资源友好性, 以构建更全面的 GUI 语义理解模型, 进一步帮助开发人员更有效地进行移动互联网的无障碍化改造, 提升检测和修复任务的效率, 促进无障碍技术的发展.

## 1.2 移动端 GUI 可访问性问题检测方法

确保移动端 GUI 对视觉障碍用户的可访问性, 不仅是实现信息平等的重要方法, 也是各国法律法规的要求. 例如, 美国的《残疾人法案》(ADA)<sup>[43]</sup>和欧洲的《欧洲无障碍法案》(EAA)<sup>[44]</sup>都对数字产品的可访问性提出了明确规定, 要求企业和组织需要确保其网站和数字产品具有可感知性、可操作性、可理解性. 因此, 检测移动端 GUI 的可访问性对开发者和企业都具有重要意义. 通过积极识别并解决可访问性问题, 企业不仅能确保符合法律法规, 避免潜在的法律风险, 还能提升产品质量, 获得竞争优势, 实现商业与社会责任的双赢.

早期针对移动端 GUI 的可访问性检测大多数都基于人工制定的启发式规则, 这种方法依赖于预先定义的可访问性标准和指南, 如万维网联盟(W3C)发布的《移动网页最佳实践》(MWBP)<sup>[45]</sup>和《网络内容可访问性指南》(WCAG)<sup>[6]</sup>. 通过检查移动应用的 GUI 是否符合这些规则识别潜在的可访问性问题. 最初的无障碍检测可以通过静态代码分析工具或在自动化测试框架下编写测试用例的方法进行. 2011 年, Android Studio 提出了一个名为 Lint<sup>[46]</sup>的静态代码检测工具, 通过对 Android 工程的源文件进行扫描, 可以检查并报告安卓项目中的可访问性问题. 在动态检测方面, 有自动化检测工具, 如 Espresso<sup>[47]</sup>. 开发人员通过运行 Espresso 检测组件是否符合 W3C 定义的可访问性问题, 需要手动指定测试用例, 并将特定的 API 嵌入到应用程序中. 2015 年, IBM 公司开发了 Mobile Accessibility Checker(MAC)<sup>[48]</sup>, 这是一个面向移动应用的无障碍检测工具, 帮助开发人员检查移动端 GUI 是否符合无障碍设计标准. MAC 可以自动检测界面组件、色彩对比度、触控目标大小、屏幕阅读器兼容性问题. MAC 的自动化检测极大提高了效率, 尤其是针对大型项目时, 可以快速发现潜在的问题. 此外, MAC 允许开发人员在应用开发的不同阶段, 实时检测无障碍问题, 并提供具体的修复建议. 该方法有助于减少开发后期的返工量, 并提升产品的无障碍质量. MAC 支持 iOS 和 Android 平台, 使开发者能在不同平台上优化其应用的无障碍体验. 但随着 GUI 设计的不断发展以及应用功能的多元化, MAC 原有的启发式规则逐渐难以满足新兴的设计和性能需求. MAC 的检测能力在面对当代 GUI 和交互设计需求时明显下降, 无法为开发者提供足够全面的无障碍优化建议.

Google 公司相继推出了两个面向安卓移动端无障碍检测的工具 ATFA (accessibility test framework for Android)<sup>[49]</sup>和 AS (accessibility scanner)<sup>[50]</sup>. ATFA 是一个用于自动化测试移动端 GUI 可访问性的库, 可以直接集成到开发者的测试代码中, 帮助开发者在应用的开发和测试阶段通过自动化方式检测无障碍问题. AS 是一个面向安卓移动设备的可访问性检测应用, 允许开发者直接在设备上对应用进行无障碍扫描, 适合于非技术用户和开发者在非自动化流程中快速检测无障碍问题. 这两个工具为安卓移动端开发提供了重要的无障碍检测辅助, 前者更适用于开发和测试阶段, 更好地在前期保证移动端 GUI 的无障碍质量, 可以大幅度减少应用上线后的修改需求. 后者更适用于开发的后期阶段, 或在产品即将发布前设计师和测试用户进行无障碍检查. 这两个基于启发式规则的可访问性检测工具虽然能覆盖常见的无障碍问题, 但在检测复杂、动态内容或语义问题上存在局限性, 它们缺乏对语义信息的深入检测. 例如, 移动端 GUI 中的图像、按钮等的用途有时难以通过工具自动识别, 难以判断移动端 GUI 中是否提供了足够的描述性文本或语义信息. 此外, 在复杂的移动应用中, 仍需手动测试覆盖无法自动检测的无障碍问题.

根据对大量安卓 App 的 GUI 进行可访问性的实证研究<sup>[51]</sup>, 研究人员发现仍有大量的移动端 GUI 由于开发人员缺乏对视障用户或老年用户无障碍的认知而导致可访问性问题. 在学术界, 研究人员不断改进对移动端 GUI 的检测方案. 基于图卷积神经网络模型提出的检测方法 ALVIN<sup>[32]</sup>, 能结合组件属性和位置关系, 通过多分类任务完成对 GUI 内可访问性问题的有效测试, 具有较高的准确率和良好

的可扩展性. Eler 等<sup>[52]</sup>提出了 Mobile Accessibility Testing(MATE)用于自动化移动应用的可访问性测试. MATE 无需依赖现有测试用例,通过自动生成测试探索应用程序行为,不再依赖开发者已有的测试套件,它不仅检测缺失的内容描述和低对比度等常见问题,还针对不同的视觉和运动障碍提供多种优化检测,例如触摸区域大小不合规、点击区域重复、可点击文本不可访问等,使可访问性检测更广泛.此外,MATE 引入状态抽象,避免重复检测相同组件,采用图模型自动构建用户界面状态,从而减少不必要的计算以提高运行效率. Chen 等<sup>[27]</sup>提出了 XBot,能高效地自动化探索用户界面.与传统工具相比,XBot 在页面覆盖率和问题检测方面性能更优.但 XBot 在应对 GUI 的动态生成和状态变化时,可能会出现冗余标记或遗漏问题,难以全面覆盖动态加载的内容和交互状态.基于启发式规则提出的无障碍检测工具通常无法适应日新月异的移动端 GUI 设计页面.检测结果的滞后性在很大程度上影响了开发人员使用这些工具的积极性.

随着深度学习技术的不断发展,尤其是基于数据驱动模型在图像和自然语言处理等领域的应用取得了显著进展.与启发式规则相比,深度学习在移动端 GUI 可访问性检测上表现出更大的潜力. Liu 等<sup>[53]</sup>提出了一个名为 OwlEye 的深度学习框架,基于卷积神经网络模型识别移动端应用 GUI 可访问性问题.这是首次将深度学习用于 GUI 无障碍问题检测,模型通过模拟人眼识别视觉异常的能力,能更准确地发现无障碍问题. OwlEye 使用梯度加权类激活映射技术(Grad-CAM)实现 GUI 显示问题的定位. OwlEye 不仅可以检测到移动端 GUI 截图中的显示问题,还可以精确定位问题区域,为开发人员提供可视化反馈,从而更高效地进行问题修复.与其他 13 种传统的机器学习和深度学习方法相比,OwlEye 在 UI 显示问题检测的召回率和精确率上分别提升了 17% 和 50%,并达到 85% 的检测精度和 84% 的召回率.同时,用户研究显示,OwlEye 在移动端 GUI 中的问题定位精确度达 90%. Salehnamadi 等<sup>[54]</sup>提出了一个高保真、场景驱动的自动化安卓无障碍自动化测试框架 Latte,该框架借助图像识别和辅助技术,如安卓的 TalkBack 和 SwitchAccess,以模拟无障碍用户的交互过程.工具会自动执行使用场景中的各项操作,并分析辅助服务在交互过程中的性能,识别在辅助服务模式下的可访问性缺陷. Latte 在多种实验和实测中显示出识别复杂问题的能力,如动态布局变化、导航循环、非标准 UI 实现等,这些问题会影响无障碍用户的正常操作,却难以通过传统的静态规则检测.与 Google 公司的 AS 可访问性检测应用相比,Latte 减少了不相关警告的数量,并提供了高保真度的使用场景分析.

表 2 列出了移动端 GUI 可访问性问题检测方法.

表 2 移动端 GUI 可访问性问题检测方法

Table 2 Methods for detecting accessibility issues in mobile GUI

方法	提出时间	主要特征
Lint <sup>[46]</sup>	2011 年	一个静态代码分析工具,可以检测 Android 项目中与无障碍相关的问题
Espresso <sup>[47]</sup>	2013 年	一个用于 Android UI 测试的框架,支持编写自动化测试用例以验证界面元素的无障碍特性
Mobile Accessibility Checker <sup>[48]</sup>	2015 年	一个面向移动应用的基于规则的无障碍检测工具
ATFA <sup>[49]</sup>	2015 年	一个用于自动化测试移动端 GUI 可访问性的库,可集成到代码中
AS <sup>[50]</sup>	2016 年	一个面向移动应用的可访问性检测应用,允许开发者和用户直接在设备上对应用进行无障碍扫描
Mobile Accessibility Testing <sup>[52]</sup>	2018 年	一个自动化面向移动应用的可访问性测试的工具,通过自动生成测试用例探索应用程序行为
OwlEye <sup>[53]</sup>	2020 年	一个深度学习框架,基于 CNN 模型识别移动端应用 GUI 可访问性问题
XBot <sup>[27]</sup>	2021 年	一个页面覆盖率和问题检测方面性能更优的可访问性检测工具,借助 Google 公司的 ATFA 工具,能高效地自动化探索用户界面
Latte <sup>[34]</sup>	2021 年	一个安卓无障碍自动化测试框架,该框架借助图像识别和辅助技术,以模拟无障碍用户的交互过程

由表 2 可见,现有的移动端 GUI 无障碍检测方法各有优势与不足.基于启发式规则的方法能快速检测常见的无障碍问题,但面对复杂、动态和语义信息不足的界面时存在局限.基于深度学习的方法

通过引入机器学习和图像识别技术, 提高了检测的准确性和覆盖范围, 但仍面临模型训练数据不足和泛化能力的挑战. 未来的研究工作可能对多模态数据进行融合, 以及利用更先进的深度学习模型, 提高对复杂无障碍问题的识别能力. 对于移动端互联网开发人员, 这些工具的进步将有助于更高效地检测无障碍问题, 并将其清晰地反馈给移动互联网开发人员.

### 1.3 移动端 GUI 布局的无障碍修复方法

GUI 布局是移动端应用程序的骨架, 决定了界面元素的组织方式和交互流程. 移动端 GUI 布局的修复在移动端无障碍中扮演至关重要的角色, GUI 布局直接影响视障用户与应用的交互方式, 优化 GUI 布局可显著提升他们的使用体验. 良好的移动端 GUI 布局包括两方面: 元素顺序的逻辑性和层级结构的清晰性. 首先, 视障用户通常依赖屏幕阅读器线性地浏览界面, 如果布局顺序混乱, 则屏幕阅读器的朗读顺序也会混乱, 导致视障用户无法建立正确的心智模型; 其次, 明确的界面层级有助于用户理解应用的功能结构, 方便视障用户导航和操作.

早期的 GUI 布局设计主要依赖设计师的经验和一系列经验法则, 例如“界面中信息不使用超过 4 种颜色”等<sup>[55]</sup>. 这些经验法则源于对人类视觉认知规律的观察和总结, 设计师通过这些原则控制界面的复杂性, 避免用户在视觉上产生混淆或认知负担. 此外, 基于网格的移动端 GUI 布局设计策略<sup>[56]</sup>是通过将页面划分为若干列和行组织界面元素, 以确保内容排列有序且一致. 这种布局方法有助于用户在不同的页面中快速定位常见元素, 特别是对视觉障碍或老年用户, 网格结构带来的内容整齐、对齐和一致性, 能有效减少他们的认知负担, 使导航更简便. 但网格布局在无障碍应用场景中也存在显著的局限性. 由于其以固定的行列比例为基础, 通常缺乏对个性化需求和动态内容的适应性. 例如, 视障用户通常依赖屏幕阅读器逐行读取界面内容, 而网格布局的固定顺序可能不符合他们的实际需求, 导致屏幕阅读器不能优先朗读关键信息. 同时, 对于低视力用户, 固定的网格布局在适配字体放大或高对比度模式时, 可能导致元素错位、重叠或信息截断, 进一步影响可读性.

Zeidler 等<sup>[57]</sup>从互联网开发人员的角度出发提出了一个名为 Auckland Layout Editor(ALE)的新型 GUI 布局编辑器, 显著简化了基于约束的布局创建和编辑过程. ALE 的一个重要特点是自动生成并管理非重叠约束, 以确保无论布局尺寸如何变化, 组件都不会互相重叠. 在布局操作中, 该工具能自动保持组件的对齐, 避免了手动调整对齐的繁琐过程, 确保了界面元素的视觉整洁. ALE 使用一种矩形网格分块技术管理布局空白区域和组件间的间距, 以确保布局的最小尺寸和最大尺寸一致且合理. 这些对用户界面的可读性和可理解性非常有益, 有利于满足视觉障碍用户的需求. UI Automator Viewer<sup>[58]</sup>是一个用于分析和检查 Android 应用程序界面布局的工具, 帮助开发和测试人员了解应用的 GUI 结构并执行可用性和无障碍性检查. 它提供了一种视觉化的方式查看应用的 GUI 层次结构, 并可显示每个组件的详细属性. Patil 等<sup>[59]</sup>对 UI Automator Viewer 工具进行改进, 使其更有效地支持移动端 Android 应用程序的无障碍性检测. 改进后的 UI Automator Viewer 引入了“捕获与播放”功能, 支持在工具的截图上执行触摸事件和按键事件. 这允许开发者模拟用户与应用程序的交互, 并自动捕获后续活动的界面. 这种功能为应用无障碍测试提供了更真实的用户交互模拟, 帮助开发者检测 GUI 的交互问题. 为帮助开发人员检测并修复视觉设计中的颜色对比问题, 工具增加了颜色对比功能. 它可以将捕获的屏幕截图转换为灰度图像, 帮助开发人员确保文本与背景之间有足够的对比度, 以便低视力或色盲用户能更轻松地访问应用. 这项功能对修复无障碍布局的可视性和可读性具有重要意义.

在传统方法的基础上, Oulasvirta 等<sup>[60]</sup>首次系统性地将组合优化方法引入到 GUI 设计领域, 将 GUI 设计问题数学化为整数规划任务. 这种方法允许识别问题类型, 分析其复杂性, 并利用已知的算法解决方案, 提供了在 GUI 设计中进行结构化和优化的有效方法. 他们提出了如何将人因相关的设计目标(如用户性能、视觉识别、运动控制等)整合为评估函数, 以帮助优化设计输出. 这种方法对如何将用户需求和交互行为纳入优化过程提供了重要的实践指导. 相比于传统基于经验法则的设计方式, 这种优化方法具有更高的灵活性, 允许直接控制设计输出, 并且能在设计工具中作为辅助设计师的交互式工具. 适应性设计方法可以将用户的特定需求纳入优化目标中, 该方法可以应用于移动端无障碍

GUI 布局设计中, 通过根据视障用户或行动受限用户的需求调整界面元素位置、大小等, 以提供个性化的无障碍体验. OR-约束(or-constraints, ORC)<sup>[61]</sup>是一种高级约束机制, 用于 GUI 布局中, 旨在提高布局的灵活性和自适应性. 它是指在布局时允许多个候选约束条件中只有一个必须被满足, 从而在调整时提供更多的排列选择. OR-约束的使用场景广泛, 例如, 在界面空间不足时, OR-约束可用于决定哪些小部件可以隐藏或被替代. OR-约束与 GUI 布局的无障碍设计密切相关, 它通过提供灵活、自适应的布局, 帮助界面更好地适应各种设备和用户需求, 特别是在优化无障碍特性方面. Jiang 等<sup>[62]</sup>提出了一个 OR-约束的自适应 GUI 布局的高效求解器 ORCSolver, ORCSolver 是首个能在近乎交互速度下解决复杂 GUI 布局的求解器, 其结合了线性约束和流布局, 从而解决了传统布局模型在处理设备多样性和不同屏幕尺寸时的限制, 有助于确保用户界面在各种移动设备上都能保持可用性和可访问性, 方便视障用户在不同屏幕条件下都能方便地使用应用程序. ORCSolver 可以在界面布局变化时(如从横屏到竖屏)自动调整组件的位置和大小, 避免界面元素重叠或超出屏幕范围. 这种功能有助于提高界面的可访问性, 确保界面元素始终清晰可见. 该求解器简化了复杂布局的管理, 使开发者能更轻松地创建无障碍友好界面, 而不需要为每种设备或屏幕设置独立的布局规范. 从而鼓励更多开发人员在设计移动应用时考虑无障碍要求, 进而提高整体用户体验.

本文团队在 GUI 无障碍化优化方面也开展了针对性研究. 针对 GUI 中组件尺寸过小、组件间隔过窄以及颜色对比度不足这 3 类可访问性问题, 采用关系图卷积神经网络(R-GCN)模型提出了修复方法, 为开发人员提供详细的属性调整策略, 有效降低了 GUI 的可访问性障碍, 显著提升了用户体验<sup>[28]</sup>. 同时, 面向视障用户在 GUI 导航中遇到的可访问性问题, 借助格式塔心理学中的接近律和相似律对 GUI 内组件进行逻辑分组, 并在分组结果的基础上重新规划导航顺序, 使导航流程更符合视障用户的操作习惯和行为感知<sup>[63]</sup>. 此外, 还针对网页端可扩展向量图形(scalable vector graphics, SVG)缺少替代文本的问题, 设计了通过特征模板和匹配规则自动生成替代文本的方案, 为提升网页可访问性提供了有力的技术支持<sup>[64]</sup>. Zhang 等<sup>[19]</sup>提出了一种创新的颜色选择方法 Iris, 是首个专门用于修复 Android 应用中颜色相关可访问性问题的自动化方法. Iris 采用一种上下文感知的修复技术, 能在保持原始 UI 设计风格一致的前提下, 修复文本对比和图像对比问题. 通过构建颜色参考数据库确保设计风格的一致性. 参考数据库包含了 9 978 个应用中没有可访问性问题的颜色对, 这些颜色已经过设计者认可, 因此确保了修复后的颜色与原始设计风格协调. 此外, Iris 通过静态分析技术准确识别需要修复的 UI 组件及其属性. 该技术克服了现有工具在修复过程中定位不准确的问题, 确保了修复的有效性和精确性.

表 3 列出了移动端 GUI 布局的无障碍修复方法.

表 3 移动端 GUI 布局的无障碍修复方法

Table 3 Accessibility repair methods for mobile GUI layouts

方法	提出时间	主要特征
ALE <sup>[57]</sup>	2013 年	一个新型 GUI 布局编辑器, 使用一种矩形网格分块技术管理布局空白区域和组件间的间距, 确保布局的最小尺寸和最大尺寸一致且合理
文献 <sup>[59]</sup>	2016 年	允许开发者模拟用户与应用程序的交互, 增加了颜色对比功能, 将捕获的屏幕截图转换为灰度图像, 帮助开发人员确保文本与背景间有足够的对比度便于修复
基于组合优化方法 <sup>[60]</sup>	2020 年	一种 GUI 设计中进行结构化和优化的有效方法, 将 GUI 设计问题数学化为整数规划任务
ORCSolver <sup>[62]</sup>	2020 年	基于 OR-约束的自适应 GUI 布局求解方法, 首个能在近于交互速度下解决复杂 GUI 布局的求解器
Iris <sup>[19]</sup>	2023 年	首个专门用于修复安卓应用中颜色相关可访问性问题的自动化方法

由表 3 可见, 现有移动端 GUI 布局修复方法各有优缺点. 早期的经验法则和基于网格的布局策略提供了直观且简单的设计框架, 但在应对个性化需求和动态内容时存在一定的局限性. 如 Auckland Layout Editor(ALE)和改进版 UI Automator Viewer 等新型工具通过自动生成和管理约束, 简化了布局创建和无障碍检测的过程, 但在处理复杂布局和动态变化时仍面临挑战. 引入组合优化方法及

OR-约束和 ORCSolver 的应用, 提高了布局设计的灵活性和自适应性, 有助于满足不同用户的无障碍需求, 但这些方法通常需要较高的计算资源和专业技能支持. 类似 Iris 的自动化颜色修复工具专注于解决特定的无障碍问题, 提供了高效的解决方案, 但可能无法涵盖所有布局缺陷. 未来的研究方向可聚焦于将人工智能与用户模型相结合, 开发出自适应的 GUI 布局修复方法, 能根据用户需求和设备特性实时优化界面布局, 同时构建大规模的无障碍布局数据集, 以提升模型的泛化能力和实用性.

## 2 未来研究方向

综合以上多方面的研究工作, 移动互联网信息无障碍的研究可以进一步拓展. 首先, 多模态融合技术的发展为无障碍设计提供了新的可能性. 目前的研究大多数集中于单一模态的信息处理(如文本或图像), 而未来则可以通过融合文本、图像、音频及用户行为数据等多模态信息, 开发更智能的无障碍检测与修复工具, 以更好地理解并优化复杂的移动界面. 从而显著增强无障碍方案的适应性, 更有效地满足多样化的用户需求. 其次, 个性化与自适应设计是未来研究的重要方向之一. 不同用户的无障碍需求存在显著差异, 未来研究可以重点开发基于深度学习和用户行为分析的个性化、自适应的无障碍解决方案. 这样的系统能根据用户的视觉障碍程度、认知能力及使用习惯, 实时优化用户界面布局、交互方式与信息呈现方式, 以更好地服务特殊需求群体, 提高系统的适应性, 从而为视障用户带来更人性化、包容性的使用体验.

人工智能与大规模数据集的结合也将成为未来研究的关键领域. 当前的深度学习模型在无障碍检测中虽性能优异, 但其效果高度依赖于高质量的大规模数据集. 因此, 未来研究需要集中构建多样化、无偏见的大规模无障碍数据集, 以提高模型的训练质量和泛化能力. 此外, 研究还需关注如何开发资源友好型的模型结构, 降低计算资源需求, 使其能在移动设备上实现实时响应, 尤其在计算资源有限的环境中, 实现更高效的无障碍支持. 同时, 新兴交互模式的无障碍研究也值得深入关注. 随着虚拟现实(VR)、增强现实(AR)等新兴技术的迅速发展, 如何确保这些复杂的交互模式同样对视障用户友好, 成为未来的挑战与机遇. 针对这些新兴交互方式的无障碍支持研究, 能确保视障用户平等地访问和体验这些前沿技术, 真正实现数字世界的普惠与包容.

此外, 人机协同优化也是无障碍设计未来的发展方向之一. 当前的无障碍设计工具在自动化方面仍存在局限, 未来可以通过设计师与人工智能的合作, 形成高效的人机协同设计模式. 人工智能能快速识别并修复无障碍问题, 而设计师则在提升设计的用户体验和合理性方面具有优势, 二者结合将极大提高无障碍设计的效率与质量. 最后, 标准化与实践推广是未来不可忽视的重要方面. 尽管已有的无障碍标准(如 WCAG)为设计提供了基本参考, 但这些标准在移动端应用中的适用性和细化程度仍需进一步提升. 未来研究应致力于制定更具针对性和系统化的无障碍标准, 涵盖新兴的交互模式, 并推动这些标准在工业界的应用与推广, 从而使开发者更易于将无障碍设计融入到开发流程中, 降低实现难度, 提升普及程度.

## 3 结 论

移动互联网信息无障碍能显著提升视障用户在数字环境中的体验, 减少他们在使用过程中遇到的障碍. 但由于开发人员在项目时间紧张、缺乏无障碍设计经验, 或者对无障碍设计的重视不足等原因, 常导致移动应用在无障碍设计上的缺失、不足或不符合用户需求的情况. 因此, 移动应用无障碍的深入研究和实践推广成为解决这些问题的有效方法, 也是当前人机交互和软件工程领域的重要应用场景之一. 同时, 针对移动应用无障碍的研究, 具有重要的社会公平价值和商业应用前景. 目前, 该问题在移动互联网开发领域仍是一个亟待解决的开放性研究课题. 本文系统地总结了该领域的研究现状与实践成果, 深入探讨了现有方法的优势和不足, 以便研究人员更好地了解移动互联网信息无障碍的最新研究进展. 移动互联网信息无障碍技术的不断发展显著改善了视障用户及其他特殊需求群体的数字体验, 但仍面临诸多挑战.

首先, 在可访问性检测方面, 传统的基于启发式规则的方法虽然在发现常见的无障碍问题上表现

良好,尤其是针对静态的界面元素,但在面对复杂的动态交互及语义理解时明显不足.近年来,深度学习技术的发展为可访问性检测带来了突破性进展,通过卷积神经网络等模型自动学习界面元素特征,提高了检测的精确度和覆盖面.特别是像 OwlEye 这样结合 Grad-CAM 可视化技术的方法,能更直观地展示检测结果,帮助开发人员快速修复问题.但深度学习模型在训练数据、泛化能力等方面仍面临挑战,模型的计算复杂度及对大规模数据的需求也在一定程度上影响了其在移动设备上的应用效率.其次,在移动端 GUI 语义表征与理解方面,本文分析了从启发式规则到深度学习模型的演变过程.传统方法通过预定义的规则集识别界面元素的语义特征,而深度学习则通过自监督学习等方法实现了对图像、文本等多模态信息的有效融合,进一步提高了对界面元素语义的理解深度.例如,LabelDroid 和 Screen2Vec 等方法通过图像特征提取与语义嵌入的结合,不仅增强了语义识别的准确性,还减轻了开发人员在标签生成方面的负担.但这些方法仍需应对动态交互、复杂语义关系及模型部署等方面的难题,尤其是在有限计算资源的移动设备上,如何实现高效而精确的界面语义理解,仍需深入探索.最后,在 GUI 布局修复方面,传统的基于经验法则与网格布局的设计方法为移动界面设计提供了基础的规范,但难以应对个性化需求及动态内容的多样化需求.现代的布局修复工具如 Auckland Layout Editor (ALE)和 UI Automator Viewer 等,通过自动化的方式生成布局约束,简化了开发过程,提高了布局的一致性和无障碍性能.基于组合优化的 OR-约束与求解器 ORCSolver 则为 GUI 布局的灵活性与适应性提供了更系统化的解决方案,能在不同屏幕条件下保证布局的一致性.但这些方法在实现过程中可能需要较高的计算资源,开发成本也较昂贵.自动化布局修复工具如 Iris 则通过上下文感知的修复方法,帮助解决颜色对比度问题,增强了低视力用户的可读性和使用体验.

综上,移动互联网信息无障碍的研究已经取得了一定的进展,并在视障用户的数字生活中发挥了重要作用,但仍面临诸多挑战与机遇.未来的研究需要结合多模态数据、人工智能、用户行为建模等先进技术,通过持续的创新和优化,推动无障碍技术的深入发展,为所有用户提供更公平和优质的数字体验,真正实现数字社会的普惠与包容.

## 参 考 文 献

- [1] 中国信息通信研究院,深圳市信息无障碍研究会.中国信息无障碍白皮书 [EB/OL]. (2019-07) [2024-11-20]. <http://m.caict.ac.cn>. (CHINA ACADEMY OF INFORMATION AND COMMUNICATIONS TECHNOLOGY, SHENZHEN INFORMATION ACCESSIBILITY RESEARCH ASSOCIATION. China Information Accessibility White Paper [EB/OL]. (2019-07)[2024-11-20]. <http://m.caict.ac.cn>.)
- [2] 中国信息无障碍产品联盟.中国互联网视障用户基本情况报告 [EB/OL]. (2016-03)[2024-11-20]. <https://www.siaa.org.cn/>. (CHINA INFORMATION ACCESSIBILITY PRODUCT ALLIANCE. Basic Situation Report of Internet Visually Impaired Users in China [EB/OL]. (2016-03)[2024-11-20]. <https://www.siaa.org.cn/>.)
- [3] 人民网.信息无障碍建设概览 [EB/OL]. (2020-05-22)[2024-11-20]. <http://wza.people.com.cn/wza2013/a/xinwensudi/2020/0521/3267.html>. (PEOPLE'S DAILY. Overview of Information Accessibility Construction [EB/OL]. (2020-05-22)[2024-11-20]. <http://wza.people.com.cn/wza2013/a/xinwensudi/2020/0521/3267.html>.)
- [4] BALLANTYNE M, JHA A, JACOBSEN A, et al. Study of Accessibility Guidelines of Mobile Applications [C]//Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia. New York: ACM, 2018: 305-315.
- [5] 万维网联盟. W3C 移动端无障碍 [EB/OL]. (2018-01)[2024-11-20]. <https://www.w3.org/WAI/standards-guidelines/mobile/zh-hans>. (WORLD WIDE WEB CONSORTIUM. W3C Mobile Accessibility [EB/OL]. (2018-01)[2024-11-20]. <https://www.w3.org/WAI/standards-guidelines/mobile/zh-hans>.)
- [6] 万维网联盟. W3C 网络内容可访问性指南 [EB/OL]. (2024-03-01)[2024-11-20]. <https://www.w3.org/WAI/standards-guidelines/wcag/>. (WORLD WIDE WEB CONSORTIUM. W3C Web Content Accessibility Guidelines [EB/OL]. (2024-03-01)[2024-11-20]. <https://www.w3.org/WAI/standards-guidelines/wcag/>.)
- [7] XU T L, WANG B S, LIU H, et al. Prevalence and Causes of Vision Loss in China from 1990 to 2019: Findings

from the Global Burden of Disease Study 2019 [J]. *The Lancet Public Health*, 2020, 5(12): e682-e691.

- [ 8 ] 工业和信息化部, 中国残疾人联合会. 中国残联关于推进信息无障碍的指导意见 [EB/OL]. (2020-09-11) [2024-11-20]. <http://www.scio.gov.cn/xwfbh/xwfbh/wqfbh/42311/44021/xgzc44027/Document/1690214/1690214.htm>. (MINISTRY OF INDUSTRY AND INFORMATION TECHNOLOGY, CHINA DISABLED PERSONS' FEDERATION. Guiding Opinions on Promoting Information Accessibility by CDPF [EB/OL]. (2020-09-11) [2024-11-20]. <http://www.scio.gov.cn/xwfbh/xwfbh/wqfbh/42311/44021/xgzc44027/Document/1690214/1690214.htm>.)
- [ 9 ] 工业和信息化部. 工业和信息化部关于印发《互联网应用适老化及无障碍改造专项行动方案》的通知 [EB/OL]. (2020-12-24) [2024-11-20]. [http://www.gov.cn/zhengce/zhengceku/2020-12/26/content\\_5573472.htm](http://www.gov.cn/zhengce/zhengceku/2020-12/26/content_5573472.htm). (MINISTRY OF INDUSTRY AND INFORMATION TECHNOLOGY. Notice on Issuing the Special Action Plan for Aging and Accessibility Transformation of Internet Applications [EB/OL]. (2020-12-24) [2024-11-20]. [http://www.gov.cn/zhengce/zhengceku/2020-12/26/content\\_5573472.htm](http://www.gov.cn/zhengce/zhengceku/2020-12/26/content_5573472.htm).)
- [10] 中华人民共和国国家发展和改革委员会. 关于印发《“十四五”公共服务规划》的通知 [EB/OL]. (2021-01-10) [2024-11-20]. [https://www.ndrc.gov.cn/xxgk/zcfb/ghwb/202201/t20220110\\_1311622.html?code=&state=123](https://www.ndrc.gov.cn/xxgk/zcfb/ghwb/202201/t20220110_1311622.html?code=&state=123). (NATIONAL DEVELOPMENT AND REFORM COMMISSION OF THE PEOPLE'S REPUBLIC OF CHINA. Notice on Issuing the “14th Five Year Plan for Public Services” [EB/OL]. (2021-01-10)[2024-11-20]. [https://www.ndrc.gov.cn/xxgk/zcfb/ghwb/202201/t20220110\\_1311622.html?code=&state=123](https://www.ndrc.gov.cn/xxgk/zcfb/ghwb/202201/t20220110_1311622.html?code=&state=123).)
- [11] 国务院. 国务院关于印发“十四五”残疾人保障和发展规划的通知 [EB/OL]. (2021-07-08)[2024-11-20]. [http://www.gov.cn/zhengce/content/2021-07/21/content\\_5626391.htm](http://www.gov.cn/zhengce/content/2021-07/21/content_5626391.htm). (STATE COUNCIL. Notice on Printing and Distributing the “14th Five-Year” Plan for the Guarantee and Development of Disabled Persons [EB/OL]. (2021-07-08) [2024-11-20]. [http://www.gov.cn/zhengce/content/2021-07/21/content\\_5626391.htm](http://www.gov.cn/zhengce/content/2021-07/21/content_5626391.htm).)
- [12] BI T T, XIA X, LO D, et al. Accessibility in Software Practice: A Practitioner's Perspective [J]. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 2022, 31(4): 1-26.
- [13] LIU H X. Some Interviews of Low-Vision Users on Using Mobile Apps. Zenodo [EB/OL]. (2022-03) [2024-11-20]. <https://doi.org/10.5281/zenodo.6331279>.
- [14] PAUL T C, ROBERT W, ALI S A, et al. Automatically Detecting Reflow Accessibility Issues in Responsive Web Pages [C]//Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering (ICSE). New York: ACM, 2024: 147-1-147-13.
- [15] FOK R, ZHONG M, ROSS A S, et al. A Large-Scale Longitudinal Analysis of Missing Label Accessibility Failures in Android Apps [C]//Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2022: 461-1-461-16.
- [16] HZ Z Y, SYED F H, SAM M. “I Tend to View Ads Almost Like a Pestilence”: On the Accessibility Implications of Mobile Ads for Blind Users [C]//Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering (ICSE). New York: ACM, 2024: 197-1-197-13.
- [17] ARUN K V, MANSUR S M H, JOSE J, et al. MotorEase: Automated Detection of Motor Impairment Accessibility Issues in Mobile App UIs [C]//Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering (ICSE). New York: ACM, 2024: 209-1-209-13.
- [18] BAJAMMAL M, MESBAH A. Semantic Web Accessibility Testing via Hierarchical Visual Analysis [C]//Proceedings of the 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE). Piscataway, NJ: IEEE, 2021: 1610-1621.
- [19] ZHANG Y X, CHEN S, FAN L L, et al. Automated and Context-Aware Repair of Color-Related Accessibility Issues for Android Apps [C]//Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering. New York: ACM, 2023: 1255-1267.
- [20] SALEHNAMADI N, MEHRALIAN F, MALEK S. Groundhog: An Automated Accessibility Crawler for Mobile Apps [C]//Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering.

- New York: ACM, 2022; 50-1-50-12.
- [21] ZHANG M R, ZHONG M, WOBROCK J O. Gally: An Automated GIF Annotation System for Visually Impaired Users [C]//Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2022; 197-1-197-16.
- [22] ISLAM M T, PORTER D E, BILLAH S M. A Probabilistic Model and Metrics for Estimating Perceived Accessibility of Desktop Applications in Keystroke-Based Non-visual Interactions [C]//Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2023; 43-1-43-20.
- [23] DAS M, McHUGH T B, PIPER A M, et al. Col1ab: Augmenting Accessibility in Synchronous Collaborative Writing for People with Vision Impairments [C]//Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2022; 196-1-196-18.
- [24] WANG R, CHEN Z, ZHANG M R, et al. Revamp: Enhancing Accessible Information Seeking Experience of Online Shopping for Blind or Low Vision Users [C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2021; 494-1-494-14.
- [25] LIU X, CARRINGTON P, CHEN X A, et al. What Makes Videos Accessible to Blind and Visually Impaired People? [C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2021; 272-1-272-14.
- [26] LIU Z, CHEN C Y, WANG J J, et al. Unblind Text Inputs: Predicting Hint-Text of Text Input in Mobile Apps via LLM [C]//Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2024; 51-1-51-20.
- [27] CHEN S, CHEN C Y, FAN L L, et al. Accessible or Not? An Empirical Investigation of Android App Accessibility [J]. *IEEE Transactions on Software Engineering*, 2021, 48(10): 3954-3968.
- [28] ZHANG M X, LIU H X, CHEN C Y, et al. AccessFixer: Enhancing GUI Accessibility for Low Vision Users with R-GCN Model [J]. *IEEE Transactions on Software Engineering*, 2024, 50(2): 173-189.
- [29] 李玉聪, 汪士钦, 张梦玺, 等. 基于 WAI-ARIA 的网页导航栏地标属性的标识方法 [J]. *吉林大学学报(理学版)*, 2024, 62(3): 697-703. (LI Y C, WANG S Q, ZHANG M X, et al. Identification Method of Landmark Attributes for Web Navigation Bar Based on WAI-ARIA [J]. *Journal of Jilin University (Science Edition)*, 2024, 62(3): 697-703.)
- [30] GLEASON C, PAVEL A, McCAMEY E, et al. Twitter A11y: A Browser Extension to Make Twitter Images Accessible [C]//Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2020; 1-12.
- [31] INTERNATIONAL TELECOMMUNICATION UNION. Measuring Digital Development: Facts and Figures 2020 [EB/OL]. (2020-12-13) [2024-11-20]. <https://www.itu.int/en/ITU-D/Statistics/Documents/facts/FactsFigures2020.pdf>.
- [32] ZHANG M X, LIU H X, SONG S N, et al. Are Your Apps Accessible? A GCN-Based Accessibility Checker for Low Vision Users [J]. *Information and Software Technology*, 2024, 174: 107518-1-107518-16.
- [33] CHEN J S, CHEN C Y, XING Z C, et al. Unblind Your Apps: Predicting Natural-Language Labels for Mobile GUI Components by Deep Learning [C]//Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering. New York: ACM, 2020; 322-334.
- [34] MEHRALIAN F, SALEHNAMEADI N, MALEK S. Data-Driven Accessibility Repair Revisited: On the Effectiveness of Generating Labels for Icons in Android Apps [C]//Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. New York: ACM, 2021; 107-118.
- [35] LI T J J, POPOWSKI L, MITCHELL T, et al. Screen2vec: Semantic Embedding of GUI Screens and GUI Components [C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2021; 578-1-578-15.
- [36] MIKOLOV T. Efficient Estimation of Word Representations in Vector Space [EB/OL]. (2013-01-13)

- [2024-11-20]. <https://arxiv.org/abs/1301.3781>.
- [37] ZHANG X Y, DE GREEF L, SWEARNGIN A, et al. Screen Recognition: Creating Accessibility Metadata for Mobile Applications from Pixels [C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2021: 275-1-275-15.
- [38] XIE M L, XING Z C, FENG S D, et al. Psychologically-Inspired, Unsupervised Inference of Perceptual Groups of GUI Widgets from GUI Images [C]//Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering. New York: ACM, 2022: 332-343.
- [39] FU J W, ZHANG X Y, WANG Y W, et al. Understanding Mobile GUI: From Pixel-Words to Screen-Sentences [J]. *Neurocomputing*, 2024, 601: 128200-1-128200-11.
- [40] YAN Y F, COOPER N, CHAPARRO O, et al. Semantic GUI Scene Learning and Video Alignment for Detecting Duplicate Video-Based Bug Reports [C]//Proceedings of the IEEE/ACM 46th International Conference on Software Engineering. New York: ACM, 2024: 232-1-232-13.
- [41] ZHAO W X, ZHOU K, LI J Y, et al. A Survey of Large Language Models [EB/OL]. (2023-03-31) [2024-11-01]. <https://arxiv.org/abs/2303.18223>.
- [42] NONG S Q, ZHU J L, WU R, et al. Mobileflow: A Multimodal LLM for Mobile GUI Agent [EB/OL]. (2024-07-15)[2024-11-10]. <https://arxiv.org/abs/2407.04346>.
- [43] BISHOP P C, JONES A J. Implementing the Americans with Disabilities Act of 1990: Assessing the Variables of Success [J]. *Public Administration Review*, 1993, 53(2): 121-128.
- [44] EUROPEAN UNION. European Accessibility Act [EB/OL]. (2019-01-01) [2024-11-20]. <https://www.europarl.europa.eu/RegData/etudes/BRIE/2017/603973>.
- [45] W3C. Mobile Web Best Practices [EB/OL]. (2007-02-01) [2024-11-19]. [https://www.w3.org/2007/02/mwbp\\_flip\\_cards.pdf](https://www.w3.org/2007/02/mwbp_flip_cards.pdf).
- [46] GOOGLE. Lint [EB/OL]. (2011-01-01)[2024-11-20]. <https://developer.android.com/studio/write/lint.html>.
- [47] GOOGLE. Espresso [EB/OL]. (2013-11-01)[2024-11-20]. <https://developer.android.com/training/testing/espresso>.
- [48] IBM. Mobile Accessibility Checker [EB/OL]. (2016-10-12)[2024-11-19]. <https://www.ibm.com/blogs/age-and-ability/2016/10/12/ibm-strengthens-mobile-app-accessibility-and-usability/>.
- [49] GOOGLE. Accessibility Test Framework [EB/OL]. (2022-03-01)[2024-11-20]. <https://github.com/google/AccessibilityTest-Framework-for-Android>.
- [50] GOOGLE. Google Accessibility Scanner [EB/OL]. (2019-04-24)[2024-11-20]. <https://support.google.com/accessibility/android/faq/6376582?hl=en>.
- [51] ROSS A S, ZHANG X, FOGARTY J, et al. Examining Image-Based Button Labeling for Accessibility in Android Apps through Large-Scale Analysis [C]//Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility. New York: ACM, 2018: 119-130.
- [52] ELER M M, ROJAS J M, GE Y, et al. Automated Accessibility Testing of Mobile Apps [C]//Proceedings of the 2018 IEEE 11th International Conference on Software Testing, Verification and Validation (ICST). Piscataway, NJ: IEEE, 2018: 116-126.
- [53] LIU Z, CHEN C Y, WANG J J, et al. Owl Eyes: Spotting UI Display Issues via Visual Understanding [C]//Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering. New York: ACM, 2020: 398-409.
- [54] SALEHNAMADI N, ALSHAYBAN A, LIN J W, et al. Latte: Use-Case and Assistive-Service Driven Automated Accessibility Testing Framework for Android [C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2021: 274-1-274-11.
- [55] GALITZ W O. *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques* [M]. Hoboken, NJ: John Wiley and Sons, 2007: 1-896.
- [56] FEINER S K. *A Grid-Based Approach to Automating Display Layout* [M]. [S. l.]: Morgan Kaufmann

Publishers Inc, 1998; 249-254.

- [57] ZEIDLER C, LUTTEROTH C, STURZLINGER W, et al. The Auckland Layout Editor: An Improved GUI Layout Specification Process [C]//Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology. New York: ACM, 2013: 343-352.
- [58] ALEXAL L. UI-Automator-Viewer [EB/OL]. (2020-10-09) [2024-11-20]. <https://github.com/alexal1/Insomniac/wiki/UI-Automator-Viewer;-What-Is-It-And-How-To-Use-It>.
- [59] PATIL N, BHOLE D, SHETE P. Enhanced UI Automator Viewer with Improved Android Accessibility Evaluation Features [C]//2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT). Piscataway, NJ: IEEE, 2016: 977-983.
- [60] OULASVIRTA A, DAYAMA N R, SHIRIPOUR M, et al. Combinatorial Optimization of Graphical User Interface Designs [J]. Proceedings of the IEEE, 2020, 108(3): 434-464.
- [61] JIANG Y, DU R F, LUTTEROTH C, et al. ORC Layout: Adaptive GUI Layout with OR-Constraints [C]//Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2019: 413-1-413-12.
- [62] JIANG Y, STUERZLINGER W, ZWICKER M, et al. ORCSolver: An Efficient Solver for Adaptive GUI Layout with OR-Constraints [C]//Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2020: 1-14.
- [63] ZHANG M X, LIU H X, ZHOU Y H, et al. Don't Confuse! Redrawing GUI Navigation Flow in Mobile Apps for Visually Impaired Users [C]//IEEE Transactions on Software Engineering. Piscataway, NJ: IEEE, 2024: 1-18.
- [64] ZHANG M X, ZHANG Y, GAO G Y, et al. Enhancing Accessibility of Web-Based SVG Buttons: An Optimization Method and Best Practices [J]. Expert Systems with Applications, 2023, 238: 121883-1-121883-17.

(责任编辑: 韩 啸)