

一种面向移动 APP 的半自动化可用性测试方法

刘伟, 刘正捷

(大连海事大学, 大连 116026)

摘要: **目的** 建立一种半自动化可用性测试方法, 为研究人员在使用传统可用性测试方法识别移动 APP 可用性问题时所需成本过高的问题提供一种解决思路。**方法** 使用数据采集工具 CAUX, 在用户执行测试任务的过程中以屏幕录制的方式自动捕获用户行为数据。利用图像对比技术从录屏数据中提取实际的任务执行序列, 然后与预期的任务执行序列进行对比, 自动提取存在可用性问题的页面和相关的用户行为数据, 并输出一份问题页面报告。根据报告内容可以快速确定可用性问题。最后以虾米音乐作为案例进行对比实验来验证方法的可行性。**结果** 使用该方法发现了传统方法难以发现的绝大多数存在可用性问题的页面及其可用性问题。**结论** 该方法通过远程自动捕获和半自动分析用户执行测试任务时的录屏数据, 可以有效发现 APP 在日常使用环境中存在的可用性问题, 从而降低了可用性测试成本, 有助于新手研究人员的使用, 并很好地推动产品的快速迭代。

关键词: 可用性测试; 移动 APP; 用户行为分析; 人机交互; 用户体验; 屏幕录制

中图分类号: TB472 **文献标识码:** A **文章编号:** 1001-3563(2022)02-0122-10

DOI: 10.19554/j.cnki.1001-3563.2022.02.016

A Semi-Automatic Usability Testing Method for Mobile APP

LIU Wei, LIU Zheng-jie

(Dalian Maritime University, Dalian 116026, China)

ABSTRACT: This paper aims to establish a semi-automated usability testing method to provide an idea for solving the problem of high cost when using traditional usability testing methods to identify the usability problems of mobile APP. The user behavior data is automatically captured in the process of users performing test tasks in the form of screen recording by a data acquisition tool CAUX. The actual task execution sequence is extracted from the recorded screen data, through image comparison technology. Then the pages and behavior data with usability problems are automatically extracted by comparing the actual and expected task execution sequence, and a report is generated. Based on the content of the report, usability problems can be quickly identified by researchers. Finally, a comparative experiment with Xiami Music as a case to verify the feasibility of the method. The results show that the majority of pages with usability problems and their usability issues that are difficult to find with traditional methods were discovered by using this method. The usability problems of APP in the daily use environment can be found by automatically capturing and semi-automatically analyzing the screen recording data when users perform test tasks, thereby reducing usability testing costs, benefiting novice researchers and promoting rapid iteration of products.

KEY WORDS: usability testing; mobile app; user behavior analysis; human-computer interaction; user experience; screen recording

可用性在 ISO 9241-11 中被定义为“特定用户在特定环境中使用产品以达到其有效性、效率和满意度

的特定目标的程度”^[1], 它是决定软件产品成败的关键因素^[2]。随着智能手机的普及, 移动 APP 逐渐覆盖

收稿日期: 2021-09-20

作者简介: 刘伟(1995—), 男, 济南人, 大连海事大学硕士生, 主攻计算机科学与技术。

通信作者: 刘正捷(1958—), 男, 大连人, 硕士, 大连海事大学教授, 主要研究方向为人机交互。

到生活中各个场景，因此提高 APP 可用性的需求不断增加^[3]。可用性测试是在实践中为了提高产品可用性而最常使用的方法之一^[4]。但是可用性测试是一项复杂且昂贵的任务，在可用性测试期间，研究人员会观察用户的行为，测量各种指标，听取用户的评论，并根据收集到的材料来确定可用性问题^[5-7]。可用性问题的发现取决于研究人员的经验，其过程没有明确的指导，并且需要花费大量的时间来进行人工的数据收集和分析，这些约束是软件开发人员在提高软件可用性时面临的最重要的挑战。另外可用性测试并不是一次性的过程，而是需要在软件迭代设计中不断重复该过程，因此在先前的研究中已经提出了轻量化分析的需求^[4]。在本项研究中，通过自动采集和分析用户在目标 APP 中执行测试任务时的录屏视频，来提取存在的可用性问题，从而帮助研究人员快速识别这些问题，提高可用性测试的效率。

1 研究现状

通过对可用性测试中的用户数据进行自动化采集或处理，可以有效降低可用性测试的成本。根据数据来源的不同可以将现有工作方法分为两类，分别是基于日志和基于视频的方法。基于日志的方法需要采集用户在执行测试任务时的埋点日志，通过挖掘异常行为模式^[8-9]或将行为路径可视化^[10]来观察和发现存在可用性问题的地方。由于大多数移动 APP 无法获取其源代码，不能像 web 应用一样通过向页面插入脚本代码的方式来获取日志数据，所以基于日志的方法常应用于 web 应用。在人机交互领域，通过录屏视频和屏幕截图来分析用户交互和 APP 可用性是一种常见的方法^[11-12]。为了方便研究人员分析录屏视频，Weichbroth 等人^[13]开发出回顾性视频数据分析工具 RVDA，研究人员可以使用该工具对视频中的交互事件和可用性问题进行人工注释，工具会自动为注释的事件标记时间戳并以 CSV 文档的形式导出，方便研究人员进行后续的分析而无需重复观看视频，但手动注释视频是一个费时的过程。Jeong 等人^[14]提出一种用户行为可视化方法，利用图像对比去除录屏

视频中重复的帧，并将剩余的屏幕图片以时间轴的形式呈现出来，使研究人员可以通过查看这些图片序列来分析用户行为，并发现其中的可用性问题。此方法可以使研究人员直观地分析用户行为，提高了分析效率，但仍需要人工查看并分析每一个图片序列。

相关研究工作表明，通过采集和分析用户执行测试任务时的日志或录屏数据，可以以较低的成本发现可用性问题。但是基于日志的方法需要研究人员具有一定的代码基础，且无法接触大多数移动 APP 的源代码，因此不能适用于所有 APP。基于视频的方法虽然解决了这一问题，但已有的方法仍需要较高的人工成本来分析数据。目前缺少一种轻量化可用性的测试方法，对移动 APP 普遍适用并能自动识别存在的可用性问题。为解决上述问题，本文基于一款数据采集工具 CAUX，以屏幕录制的方式自动捕获用户执行测试任务时的行为数据，然后采用图像对比技术和序列对比算法对数据进行分析，自动提取出存在可用性问题的页面和相关的用户行为数据，并将结果生成一份问题页面报告。研究人员根据报告内容可以快速识别出可用性问题。

2 半自动化可用性测试方法

本文提出方法共分为 4 个步骤：预期任务序列及基准图片定义、用户行为捕获、用户行为序列提取和可用性问题检测。首先，研究人员根据完成每个测试任务所必须经过的页面名称定义预期的任务执行序列，并截取 APP 中的关键页面图片作为基准图片集；其次，使用 CAUX 工具自动采集用户执行每个测试任务时的录屏视频；再次，利用图像对比技术将录屏视频中的帧图片与基准图片集进行对比，提取出由基准图片名称表示的用户实际执行任务的行为序列；最后，根据所提出的偏差计算方法对每个任务的用户行为序列和预期任务序列进行对比分析，自动提取出与可用性问题相关的页面和用户行为数据，并形成问题页面报告，研究人员根据报告内容可以确定可用性问题，见图 1。

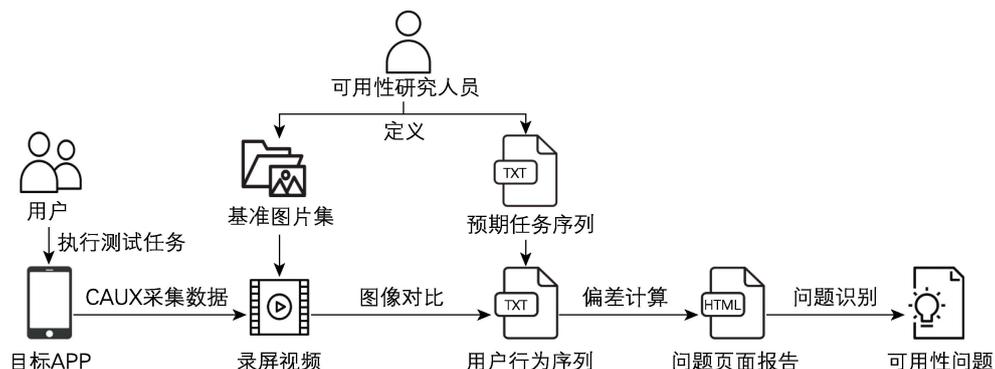


图 1 方法的体系结构

Fig.1 Architecture of the proposed method

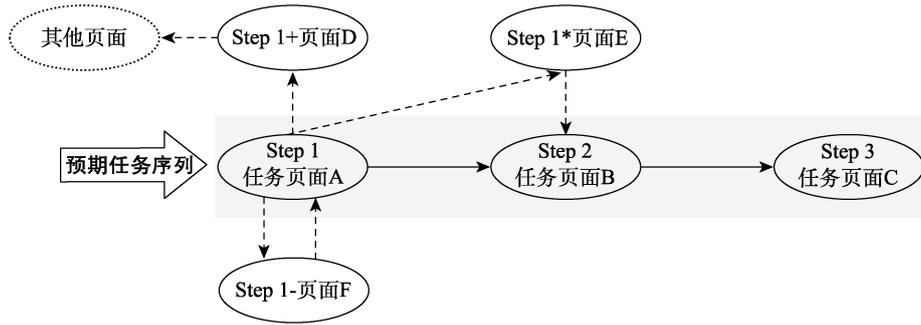


图2 可用性测试任务示例
Fig.2 Example of a usability testing task



图3 存储预期任务序列的文件示例
Fig.3 Example of a file storing the expected task sequence

2.1 预期任务序列及基准图片定义

预期任务序列由 2 部分组成,即完成任务所必须经过的任务页面名称(如图 2 中的页面 A, B 和 C)和预期用户在每个任务页面中停留时长的基准值,见图 2。页面的预期停留时长由研究人员根据在页面中用户需要执行的操作和常见页面类型的一般规则来定义,例如人为因素指南规定了页面响应时间的上限,以避免用户失去耐心^[15]。最后将定义好的预期任务序列写在一个 txt 文件中,每一行由任务页面的“预期停留时长(秒)+页面名称”组成,文件命名为“task+任务编号”,见图 3。

通过将录屏视频和基准图片集进行对比,可以提取出由基准图片标识的用户行为序列。基准图片较为集中的图片包括以下 2 类页面的截屏:(1)预期任务序列中的任务页面;(2)在每个任务页面中用户可能会直接进入的非任务页面,例如图 2 中的页面 D、E 和 F。截取的基准图片为进入上述页面后第一个状态的屏幕截图。由于页面中的部分内容会经常发生变化,所以为了提高用户行为识别的准确度,需要为每张基准图片定义唯一代表该页面的矩形固定区域(每次进入该页面,该区域的元素固定不变),在进行图像对比时,只对图片中的固定区域进行对比,见图 4。

2.2 用户行为捕获

CAUX 是一种数据采集工具,由客户端和服务端 2 部分组成^[16]。客户端安装运行在用户手机上,但当前所能采集的数据仅包括网络状态、光照强度等情



图4 基准图片及固定区域示例
Fig.4 Reference picture and example of fixed area

境数据,为 CAUX 增加了录屏功能,从而使其可以采集 APP 中的用户行为数据。使用 CAUX 采集用户数据可以分为 4 个步骤,见图 5。

1) 研究人员设置指令。研究人员将 APP 名称和要采集的数据类型写进指令文件,然后将指令文件上传到服务器。每条指令包含触发条件和采集动作。以虾米音乐为例,一条指令可以设置触发条件为虾米音乐前台运行,采集动作为开启录屏和采集电量信息。

2) 用户查看并执行任务。用户查看测试任务,在 CAUX 中输入自己的编号并启动 CAUX,进入目标 APP 执行任务。

3) CAUX 客户端解析指令文件,自动采集数据。CAUX 启动后会从服务器下载并解析指令文件,循环监听情境信息,若满足指令文件中的触发条件,则执行采集动作。

4) 数据上传。采集到的录屏视频和情境数据文件均以用户编号和时间戳命名,并暂存在本地存储器中。当用户确认数据未涉及隐私后则通过 CAUX 通

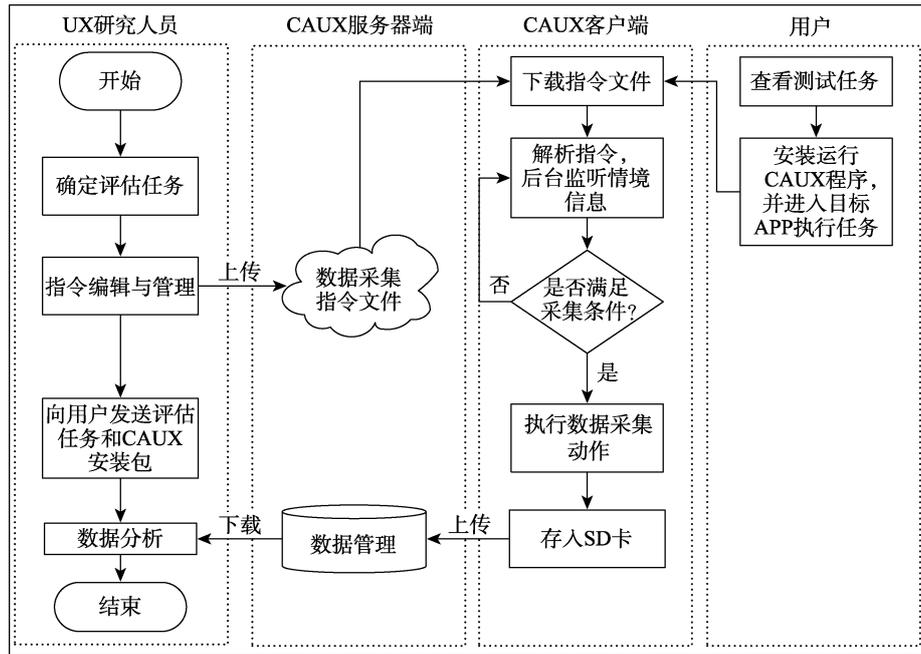


图 5 使用 CAUX 采集用户数据的流程
Fig.5 Flow of using CAUX to collect user data

信模块上传至服务器。

2.3 用户行为序列提取

为了从录屏视频中提取用户行为序列, 利用图像对比技术将视频的帧序列与基准图片集中进行对比, 映射为由基准图片标识的用户行为序列。每个视频的处理过程分为 3 个步骤:

1) 预处理。首先将视频读取为帧序列, 并依次将前后两帧进行整张图片的对比, 去除相邻且重复的帧, 每一组重复帧只保留第一帧。

2) 匹配基准图片, 并生成用户行为序列。将剩余的每一帧图片与基准图片集中的图片进行固定区域的对比(对比时先获取基准图片中固定区域的对角坐标, 然后按坐标值截取两张图片中的对应区域进行对比), 匹配到与该帧相同的基准图片后, 保存该帧的时间戳和对应基准图片的名称, 并输出到日志文件中。

3) 日志去重。最后由于某些页面中存在部分动态变化的元素(如动图), 导致用户即使一直停留在这些页面未进行操作, 视频中该页面的帧图片也会一直变化。因此第一步的预处理工作无法筛除掉这部分图片, 再经过固定区域的对比, 会导致生成的日志文件中有同一页面重复出现多条日志的情况, 因此仅保留第一条最早的日志。

其中, 使用的图像对比算法为感知哈希算法。由于视频名中包含该视频的时间戳, 先将视频的时间戳赋值给第一帧, 然后通过计算每一帧在视频中经过的毫秒值, 可以获得每一帧图片的时间戳。

每个任务视频经过图像对比都会生成一个 txt 格式的日志文件, 并以“task+任务编号+用户编号”进

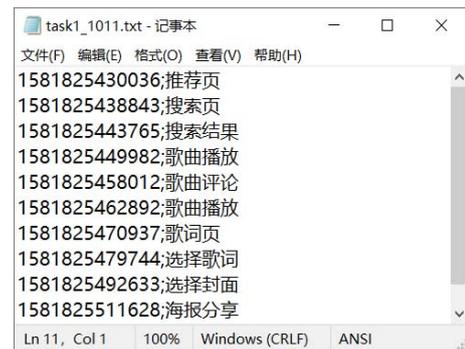


图 6 存储用户行为序列的文件示例
Fig.6 Example of a file storing user behavior sequence

行命名。日志文件中的每条日志由“进入某个页面的时间戳+该页面的名称”组成, 这些日志条目构成了用户执行该任务的行为序列, 见图 6。

2.4 可用性问题检测

由于可用性问题的存在, 导致用户实际执行任务时无法按照预期的任务路径和任务时间来完成任务, 可能会出现逻辑和时间上的偏差。本文提出 2 种偏差计算方法来对用户行为序列和预期任务序列进行对比和计算, 识别出用户行为序列中的 2 种偏差, 并提取相关的问题页面和操作数据。最后将计算的结果生成一份问题页面报告, 研究人员通过报告可以较为容易地发现其中的可用性问题。

2.4.1 提取问题页面

逻辑偏差计算。将用户在任务页面中执行的每个不符合预期的页面跳转视为一次逻辑偏差。在某个任务页面中, 将所有用户出现的逻辑偏差总数定义为该页面的逻辑偏差值。同时将统计每个任务页面中出现

逻辑偏差的人数、进入的非任务页面及进入的次数。通过逻辑偏差计算来提取所有出现逻辑偏差的页面和相关数据。每个任务的逻辑偏差计算算法见算法1。

算法1: 逻辑偏差计算算法。

输入: 某任务预期任务序列文件路径 path1, 存储该任务所有用户行为序列文件的文件夹路径 path2。

输出: 存储该任务逻辑偏差对象的集合 LDS。

步骤1: 读取预期任务的序列文件, 为文件的每一行创建一个页面对象, 用来保存该行中的页面名称和预期停留时长, 并创建集合 S 来存储这些页面对象, 然后进入步骤2。

步骤2: 从 S 中获取前两个页面对象 pre 和 next, 并创建一个逻辑偏差对象 logicDev, 将 pre 赋值给 logicDev 的问题页面变量, 然后进入步骤3。

步骤3: 按顺序读取一个用户行为序列文件, 如果读取成功, 则为文件的每一行创建一个页面对象, 用来保存该行中的页面名称和时间戳, 并创建集合 U 来存储这些页面对象, 然后进入步骤4; 否则进入步骤5。

步骤4: 从 U 中获取与 pre 相邻的下一个页面对象, 如果获取成功, 将非 next 的页面添加到 logicDev 的非任务页面变量中, 使 logicDev 的逻辑偏差用户数变量+1, 然后销毁集合 U, 进入步骤3; 否则直接销毁集合 U, 进入步骤3。

步骤5: 如果 logicDev 的逻辑偏差用户数大于0, 则将其添加到集合 LDS 中; 否则清空 logicDev 对象, 进入步骤6。

步骤6: 从 S 中获取下一个页面对象, 如果获取成功则将该页面对象作为新的 next, 并将原来的 next 作为 pre, 然后进入步骤3; 否则结束运算。

其中将 logicDev 定义为一个逻辑偏差对象, 其包含4个成员变量: 出现逻辑偏差的页面、该页面中出现逻辑偏差的人数、逻辑偏差值及用户从该页面进入的非任务页面集合。

计算逻辑偏差的示例, 见图7。页面A中的3名用户执行的跳转操作全都符合预期路径, 未出现逻辑偏差; 页面B中有2名用户出现了3次错误操作, 因此页面B存在逻辑偏差, 出现偏差的用户数为2, 逻辑偏差值为3, 进入的错误页面为D和E。

时间偏差计算。如果用户从一个任务页面进入到

下一个任务页面所经历的时间超过了为前一个页面定义的预期停留时长, 则视为一次时间偏差, 见图8。

对于一个存在时间偏差的任务页面, 应统计在该页面出现时间偏差的用户数, 并计算这些用户进入下一个任务页面的平均跳转时长, 将平均跳转时长与预期停留时长的差值定义为时间偏差值。通过时间偏差计算, 来提取所有出现时间偏差的页面和相关数据。每个任务的时间偏差计算过程见算法2。

算法2: 时间偏差计算算法。

输入: 某任务预期任务序列文件路径 path1, 存储该任务所有的用户行为序列文件的文件夹路径 path2。

输出: 存储该任务时间偏差对象的集合 TDS。

步骤1: 读取预期任务序列文件, 为文件的每一行创建一个页面对象, 用来保存该行中的页面名称和预期停留时长, 并创建集合 S 来存储这些页面对象, 然后进入步骤2。

步骤2: 遍历 S, 为 S 中每个页面对象创建一个时间偏差对象 timeDev, 并使 timeDev 的问题页面变量存储对应的页面对象, 将所有 timeDev 存储到集合 TDS 中, 然后进入步骤3。

步骤3: 按顺序读取一个用户行为序列文件, 如果读取成功, 为文件的每一行创建一个页面对象, 用来保存该行中的页面名称和时间戳, 并创建集合 U 来存储这些页面对象, 然后进入步骤4; 否则进入步骤9。

任务1-任务页面A		任务1-任务页面B	
预期路径:	(A, B)	预期路径:	(B, C)
出现偏差的用户数:	0	出现偏差的用户数:	2
逻辑偏差值(L):	0	逻辑偏差值(L):	3
进入的错误页面:	Null	进入的错误页面:	D(1), E(2)

图7 逻辑偏差计算结果示例

Fig.7 Example of logical deviation calculation result

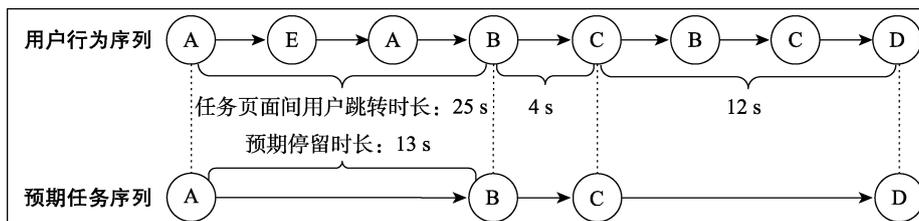


图8 用户在任务页面间的跳转时长示例

Fig.8 Example of the duration of the user's jump between task pages

步骤 4: 从 S 中获取前两个页面对象 pre 和 next, 令 mark=-1, 然后进入步骤 5。

步骤 5: 如果 pre 的预期停留时长大于 0, 进入步骤 6; 否则进入步骤 8。

步骤 6: 在 U 中从元素序号 mark+1 开始, 按顺序查找第一次出现且与 pre 和 next 页面名称相同的页面对象, 如果两者都找到, 则将两个页面对象的时间戳赋值给 Time1 和 Time2, 并将与 next 页面名称相同的页面对象的元素序号赋值给 mark, 并进入步骤 7; 否则进入步骤 8。

步骤 7: 计算 Time2-Time1 的值, 如果大于 pre 的预期停留时长, 则按照 pre 在 S 中的元素序号从 TDS 中找到相同序号的 timeDev, 并对其进行如下操作。将出现时间偏差的人数加 1; 将页面跳转的总时长加上 Time2-Time1 的值, 然后进入步骤 8; 否则直接进入步骤 8。

步骤 8: 从 S 中获取下一个页面对象, 如果获取成功则将该页面对象作为新的 next, 将原来的 next 作为 pre, 并进入步骤 5; 否则销毁集合 U, 进入步骤 3。

步骤 9: 遍历 TDS 中每个 timeDev, 如果 timeDev 的出现偏差用户数大于 0, 则将其页面跳转总时间除以出现时间偏差人数的值作为平均跳转时长, 并将时间偏差值赋值为平均跳转时长减问题页面的预期停留时长; 否则删除这个 timeDev。

将 timeDev 定义为一个时间偏差对象, 其包含 4 个成员变量: 问题页面、该页面出现时间偏差的用户数、出现时间偏差的用户在该页面的跳转总时长及时

间偏差值。

计算时间偏差的示例: 用户从页面 C 进入页面 F 的过程中未出现时间偏差; 从页面 F 进入页面 G 的过程中, 有 6 名用户超出了页面 F 的预期停留时长, 则认为这 6 名用户的操作出现了时间偏差, 并统计他们的总耗时为 90 s, 从而计算出平均跳转时长为 15 s, 时间偏差值为平均跳转时长减预期停留时长, 即为 9, 见图 9。

问题页面报告。将偏差计算的结果形成一份 HTML 文档格式的问题页面报告, 便于研究人员查看和分析其中的可用性问题。报告中每一行都显示一个存在可用性问题的页面, 将可用性问题表示为“存在可用性问题的页面和对应页面图片+存在的偏差类型+出现偏差的用户数+偏差值+问题描述”, 并允许按任务号或偏差类型对问题进行筛选, 见图 10。

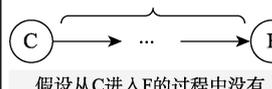
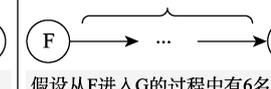
任务1-任务页面C	任务1-人物页面F
中间包含0个或多个页面 	中间包含0个或多个页面 
假设从C进入F的过程中没有用户的超出预期跳转时长	假设从F进入G的过程中有6名用户超出预期跳转时长, 且总耗时90 s
预期路径: (C, F)	预期路径: (F, G)
预期停留时长: 6 s	预期停留时长: 6 s
出现偏差的用户数: 0	出现偏差的用户数: 6
时间偏差值: 0	时间偏差值: 9

图 9 时间偏差计算结果示例

Fig.9 Example of time deviation calculation result

任务5	页面名称	页面图片	全选	出现偏差的用户数	偏差值	问题描述
5	个人中心		逻辑偏差	7	10	该页面中 7 名用户出现 10 次错误跳转, 未进入预期的 定时关闭 页面, 进入的错误页面为: 设置(8) 歌曲播放(2)
5	个人中心		时间偏差	3	11	该页面中 3 名用户未在预期时间内进入 定时关闭 页面, 平均每名用户多花费了 11 秒

图 10 问题页面报告示例

Fig.10 Example of a problem page report

2.4.2 确认可用性问题

通过两种偏差计算方法提取出来的页面,表示其存在用户操作错误和操作时间过长的的问题,会影响用户完成任务的有效性和效率^[17],预示着可用性问题的存在。

通过查看报告中的问题描述,可以大致判断问题页面中出现可用性问题的部分。因此可以借助一些公认的可用性准则(比如 Nilsen 十大可用性准则^[18])来仔细分析整个问题页面,其重点是在问题页面中用户应该进入的下一个任务页面和进入的错误页面的入口,从而判断其是否满足可用性准则,以确定可用性问题的存在。例如对于通过计算逻辑偏差提取出来的页面,应该关注其页面元素设计是否符合用户的认知或满足一致性原则,尤其是所进入的错误页面的入口,是否会误导用户或引起用户的误操作;通过计算时间偏差提取出来的页面,应该关注页面内容和元素的布局,判断是否会因为页面内容过于复杂而造成用户的认知负担,或由于入口不明显导致用户需要花费较长时间去查找或尝试,而无法顺利地进入目标页面。

3 案例研究

选取虾米音乐 APP 作为实验平台进行了传统的可用性测试。测试的同时使用 CAUX 工具收集数据,并使用方法对数据进行自动分析。最后通过对比和分析来验证所提出方法的有效性。

3.1 用户招募

根据艾媒咨询 2019 年至 2020 年中国手机音乐客户端的市场研究报告发现,在虾米音乐的用户中,30 岁以下的青年用户占比超过 70%,因此招募了 20 名年龄在 16~28 岁、有音乐类 APP 使用经验的用户参与了本次实验,男女比例为 1:1。

3.2 可用性测试任务

实验开始前为虾米音乐制定了 7 个测试任务并给出了每个任务的详细描述,见表 1。

3.3 传统可用性测试

由于测试参与者分布在不同地区,且时间难以统一,无法将他们邀请到实验室中进行测试,因此本次实验通过远程视频会议的方式来进行可用性测试。3 名测试人员分别作为主持人(1 名)和观察者(2 名)参与了测试。主持人负责组织会议并给用户发放任务,观察者需要记录用户任务时间,观察用户的行为和表情,并对发现的问题进行记录,同时也要记录用户的反馈意见。测试完成后,研究人员会根据观察和记录的情况对参与者进行访谈,对发现的可用性问题进行进一步确认并了解问题发生的原因。记录了测试过程中出现的所有问题和反馈,并对数据进行分析 and 汇总,从 11 个页面中提取出 12 个可用性问题的存在,见图 11。

3.4 基于屏幕录制的可用性测试

3.4.1 任务序列和基准图片集

按照上文中的方法为 7 个任务定义了 7 个预期任务序列和 69 张基准图片。为了提高用户行为识别的准确度,为 20 名用户出现的 3 种分辨率的手机屏幕各提供了一组基准图片集。

3.4.2 数据采集

实验开始前,需要远程指导用户完成 CAUX 的安装和使用,并为每位用户分配一个用户 ID。规定用户每完成一个任务,需要关闭虾米音乐,然后再执行下一个任务。当用户进入虾米音乐时,CAUX 会在后台根据编写的指令采集数据,见表 2。采集的数据会暂存在本地存储器,当用户完成所有任务且确认内容不涉及隐私后,再将数据通过 CAUX 的通信模块上传至服务器。

3.4.3 数据分析

本次实验共收集到 20 名用户执行测试任务时的 140 段录屏视频。采用基于图片对比的自动化数据分析方法对视频进行分析,分析步骤如下。

第一步:用户行为序列提取。根据视频文件名中

表 1 可用性测试任务
Tab.1 Usability test tasks

ID	任务名	任务描述
1	制作歌曲海报	您想搜索一首自己喜欢的歌曲,然后将歌词制作成海报分享到微信朋友圈
2	收听虾米电台	您想收听虾米电台,并为打开电台后听到的第一首歌曲留言
3	调整播放列表	您想播放“每日 30 首”中的全部歌曲,然后在当前音乐播放列表中调整播放顺序(将第 4 首歌放到第 2 首的位置)
4	批量导入歌曲	您想搜索古风相关的歌单,将前 6 首歌曲添加到一个自己新创建的歌单中
5	定时播放歌曲	假设您现在想要午睡,在睡觉前想播放任务 4 中创建的歌单,并设置 15 分钟后自动停止播放
6	查找艺人	您想了解朴树这位歌手,包括他的个人介绍和代表歌曲,并下载几首他的歌曲
7	创建房间	您想创建一个分享音乐的房间(房间),并将自己已经建立的歌单作为播放列表

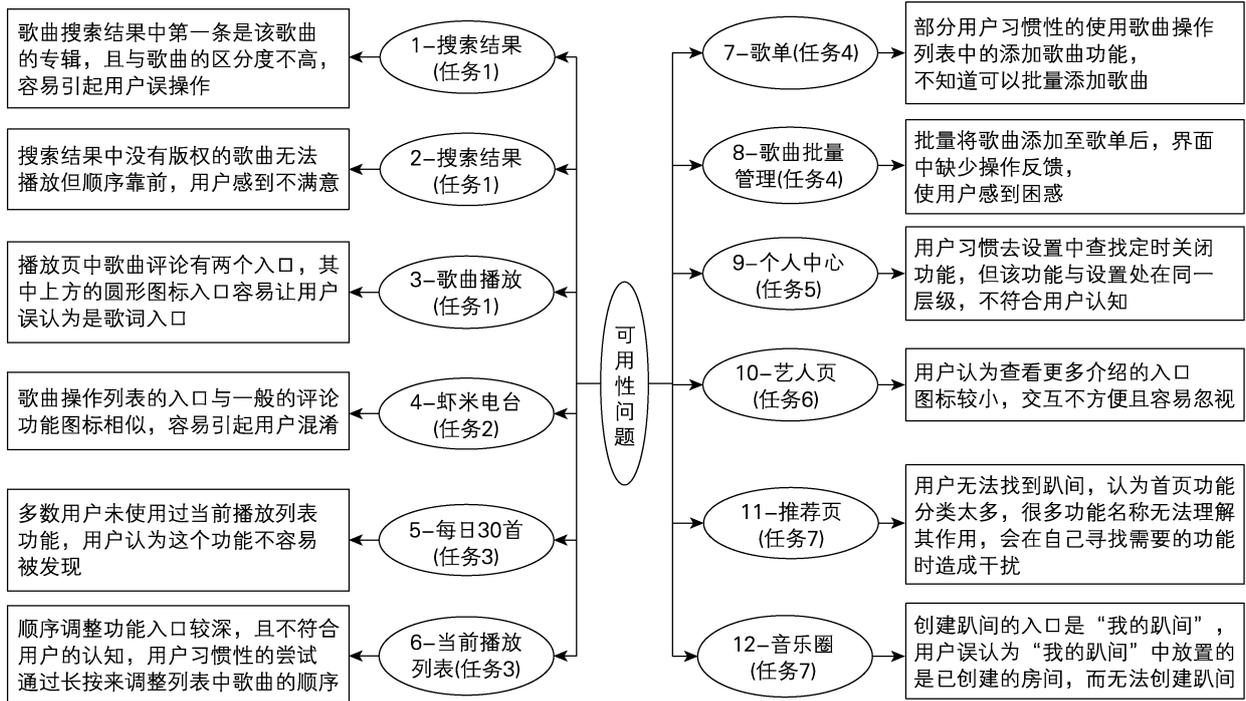


Fig.11 Usability problems found by traditional methods

表 2 数据采集指令

Tab.2 Data collection instruction

触发条件 (IF)	数据采集 (THEN)
虾米音乐开始运行	记录此时的时间、电量、联网方式，并开启录屏功能
虾米音乐关闭	记录此时的时间、电量、联网方式，并关闭录屏功能

的任务编号将视频分组，基于图像对比算法将录屏视频与基准图片集进行对比，得到 140 个用户行为序列文件。

第二步：识别问题页面。对比每个任务的预期任务序列和用户行为序列，利用上文中提出的两种偏差计算算法，计算每个任务页面的偏差值。筛选掉个别用户出现偏差的页面，提取出 10 个存在逻辑偏差和 7 个存在时间偏差的问题页面，见表 3—4。完成偏差计算后，为问题页面匹配相应的页面图片，并输出 HTML 格式的问题页面报告。

第三步：确定可用性问题。根据上文提出的方法，进一步分析问题页面报告中的偏差计算结果，以识别可用性问题。

通过上述三步，来完成用户数据分析，最终在页面中发现了 10 个可用性问题，见图 12。

3.5 实验结果分析

从提取的问题页面及可用性问题、偏差值与可用性问题严重性的关系、研究成本这 3 个方面对实验结果进行分析，以验证所提出方法的有效性。

3.5.1 问题页面及可用性问题

由上文可知，使用传统方法共发现 11 个页面中

的 12 个可用性问题，而使用本文提出的方法识别出了 10 个页面中的 10 个可用性问题，并且在本文所提出的方法中，用户进入的错误页面和操作时长也都在可用性问题报告中，并提到了导致出现可用性问题的原因。传统方法发现的另外 2 个可用性问题为问题 2 和 10，是用户主动反馈的与满意度相关的可用性问题，由于本文提出的方法目前无法采集和分析主观的数据，所以未发现这两个问题。

由此可知，本文所提出的方法可以有效地提取存在可用性问题的页面，并且根据本文给出的问题页面报告可以看出，对存在可用性问题的页面进行分析，可以识别出除了与满意度相关问题之外的大多数可用性问题。

3.5.2 偏差值与可用性问题的严重性

邀请 3 名可用性领域的专家，根据广泛使用的 Dumas 提出的严重性等级来划分标准^[19]，将使用本文提出的方法所得出的可用性问题分为一般、中等、重要和严重 4 个等级，然后列出问题页面的偏差用户数和偏差值，见表 5。

通过表中列出的数据可以明显发现，随着在页面中出现偏差的用户数和偏差值的增加，问题的严重性等级也逐渐升高。因此可以根据出现偏差的用户数和

偏差值的大小来大致判断页面中可用性问题的严重性等级。

3.5.3 研究成本

时间成本：统计研究人员在实验中的用时，见图13。传统方法需要研究人员参加每一场测试且每场测

试只允许1名用户参加，此外需要人工对数据进行记录和分析，共耗时45.5h。CAUX工具采用C/S模式，支持多用户同时参与测试，并可以对所采集的数据进行半自动化的分析，共耗时33h，比传统方法节约28%的时间。另外每增加一名用户，使用传统方法的

表3 逻辑偏差计算结果
Tab.3 Logical deviation calculation results

任务	页面	出现偏差的用户数	逻辑偏差值	进入的错误页面
1	搜索结果	3	3	音乐专辑(3)
1	歌曲播放	5	8	歌曲评论(5), 歌曲操作(2), 歌曲信息(1)
2	虾米电台	6	7	歌曲操作(7)
3	每日30首	8	24	歌曲操作(14) 批量管理(10)
3	当前播放列表	14	21	移除列表歌曲(12), 歌曲播放(8), 歌曲信息(1)
4	歌单	8	30	歌曲操作(27), 歌曲播放(3)
4	歌曲批量管理	4	4	歌单(4)
5	个人中心	7	10	设置(8), 歌曲播放(2)
7	推荐页	16	34	我的音乐(11), 搜索(9), 视频(5), 乐库(5), 个人中心(4)
7	音乐圈7	11	19	粉丝趴(11), 热门趴间(8)

表4 时间偏差计算结果
Tab.4 Time deviation calculation results

任务	页面	需要进入的页面	出现偏差的用户数	时间偏差值(结果取整)
3	每日30首	歌曲播放	8	9
3	当前播放列表	列表管理	12	14
4	歌单	批量管理	7	8
4	批量管理	加入到歌单	2	4
5	个人中心	定时关闭	3	11
7	推荐页	趴间	11	10
7	音乐圈	我的趴间	10	12

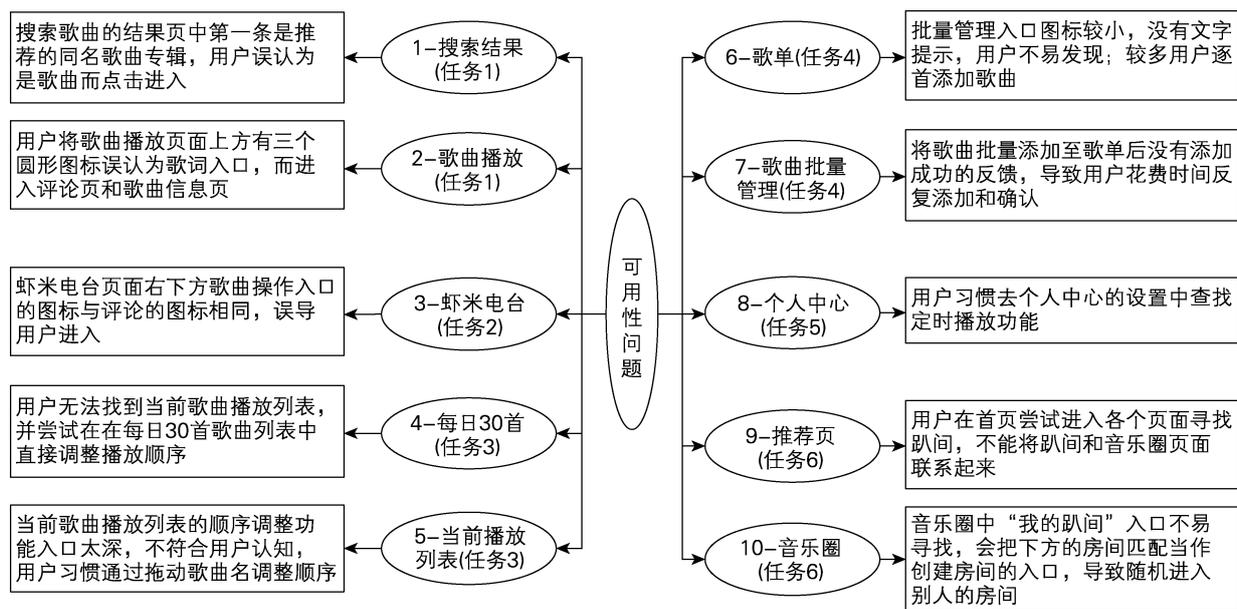


图12 使用本文提出方法发现的 usability 问题
Fig.12 Use the method proposed in this paper to find the usability problems

表 5 可用性问题的严重性等级分类
Tab.5 Severity level classification of usability problems

问题编号	问题页面	严重性等级	逻辑偏差用户数/逻辑偏差值	时间偏差用户数/时间偏差值
1	搜索结果	一般	3/3	Null
2	歌曲播放	一般	5/8	Null
3	虾米电台	一般	6/7	Null
7	歌曲批量管理	中等	4/4	2/4
8	个人中心	中等	7/10	3/11
6	歌单	中等	8/30	7/8
4	每日 30 首	中等	8/24	8/9
10	音乐圈	重要	11/19	10/12
5	当前播放列表	重要	14/21	12/14
9	推荐页	重要	16/34	11/10

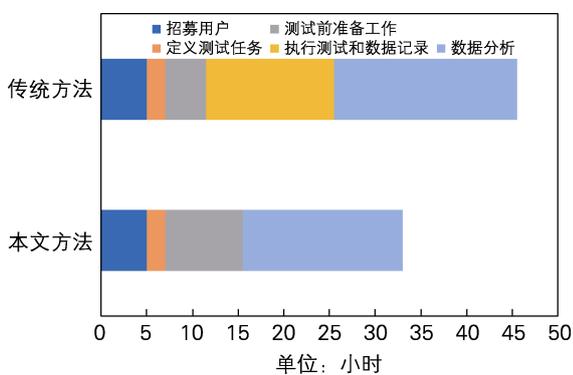


图 13 研究人员在两种方法中的用时对比
Fig.13 Comparison of researcher's time between two methods

研究人员都需要平均花费 1.9 h 去参与测试和分析数据，而使用本文提出的方法，研究人员仅需要平均花费 1.1 h 来进行测试前指导和半自动化的数据分析工作即可。因此，对于用户和任务数量较多的可用性测试，本方法较传统方法可以有效地降低时间成本。但由于使用本方法在对一款 APP 进行可用性测试前，必须花费时间进行基准图片的采集和工具安装调试等准备工作，所以对于用户和任务数量较少的小规模可用性测试，本方法不具备时间成本的优势。

人力和物力成本：一场传统的可用性测试需要 3 名测试人员和 1 名用户参与，如果是远程测试则需要 4 台笔记本和 1 部智能手机；若在实验室中进行测试则需要 1 个带有测试间和观察间的可用性实验室、1~3 台电脑、1 台摄像机和 1 部智能手机，并需要用户抽出较多时间花费在实验室中。而本方法仅需要每名用户使用 1 部智能手机完成测试，以及 1 名测试人员使用 1 台电脑进行测试准备和数据分析即可。因此本文所提出的方法有效节省了人力和物力，避免了资源浪费。

4 结语

本文提出了一种新的面向移动 APP 的轻量化可

用性测试方法，基于一款情境感知工具，以屏幕录制的方式自动采集用户行为数据，并结合图像对比技术和序列对比算法对数据进行半自动化分析，从而以较低的成本识别可用性测试问题。本次实验招募了 20 名用户并使用一款真实的 APP 进行了实验，实验结果表明，本方法不仅可以准确有效的识别可用性测试问题和划分出问题的严重性等级，并且当测试产生的数据量较大时，本方法较传统方法具有明显的成本优势。此外，预期该方法会对移动 APP 中的用户行为研究工作具有一定的参考价值。

但本文工作也存在几点不足：首先由于所提出的方法在可用性测试前需要进行基准图片的采集等准备工作，所以针对用户和任务数量较少的小规模测试，本文所提出的方法较传统方法花费的时间较多；其次无法对用户的主观数据进行收集，存在一定的局限性。后续将会进一步探索实现该方法的自动化。

参考文献：

- [1] BEVAN N, CARTER J, EARTHY J, et al. New ISO Standards for Usability, Usability Reports and Usability Measures[C]. Toronto, Canada: International Conference on Human-computer Interaction. Springer, Cham, 2016.
- [2] WEICHBROTH P. Delivering Usability in IT Products: Empirical Lessons from the Field[J]. International Journal of Software Engineering and Knowledge Engineering, 2018, 28(7): 1027-1045.
- [3] INAL Y, HACALOGLU T. Users' Behavioral Strategies Toward Mobile App Problems: Fight or Flight[C]. Trondheim, NORWAY: Conference on E-Business, E-Services and E-Society, Springer, Cham, 2019.
- [4] FOLSTAD A, LAW E, HORNBK K. Analysis in Practical Usability Evaluation: a Survey Study[C]. Austin Texas USA: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2012.

(下转第 144 页)