

Virtual Fitting System Based on OOTDiffusion and UniApp Technology

Bingbing Li

Xi'an University of Technology, Xi'an, Shaanxi, 710600, China

Abstract

Aiming at the common problems of virtual fitting technology in clothing e-commerce field, such as strong device dependence, insufficient cross-platform compatibility and low operation efficiency, a lightweight cross-platform virtual fitting system based on OOTDiffusion model and UniApp framework was designed and implemented. The hybrid rendering mechanism and conditional compilation technology of UniApp are adopted to complete the multi-terminal adaptation of iOS/Android/Web/ applet. Deploying OOTDiffusion diffusion model, through layered attention mechanism and physical simulation engine, the generated results are realistic. Optimize the reasoning process, optimize the average reasoning delay of the mobile terminal to 28.7 seconds, and control the peak memory occupation within 200MB. The system test shows that the second response can be stably realized on mainstream low-end devices, and the generated fitting images are significantly better than traditional schemes such as VITON-HD in terms of texture detail retention and human body adaptability.

Keywords

virtual fitting; UniApp; OOTDiffusion; Cross-platform; Lightweight Deployment

基于 OOTDiffusion 与 UniApp 技术的虚拟试衣系统

李兵兵

西安工程大学, 中国·陕西 西安 710600

摘要

针对服装电商领域虚拟试衣技术普遍存在的设备依赖性强、跨平台兼容性不足及运行效率低下等问题,设计并实现了基于 OOTDiffusion 模型与 UniApp 框架的轻量化跨平台虚拟试衣系统。采用 UniApp 的混合渲染机制和条件编译技术,完成 iOS/Android/Web/小程序多端适配。部署 OOTDiffusion 扩散模型,通过分层注意力机制与布料物理仿真引擎,生成结果的真实感强。优化推理过程,移动端平均推理延迟优化至 28.7 秒,峰值内存占用控制在 200MB 以内。系统测试表明,在主流中低端设备上可稳定实现秒级响应,生成的试衣图像在纹理细节保留度与人体适配性方面显著优于 VITON-HD 等传统方案。

关键词

虚拟试衣; UniApp; OOTDiffusion; 跨平台; 轻量化部署

1 引言

随着服装行业与电子商务的迅猛发展,消费者对线上购衣的高效性、精准性等需求持续提升 [1]。虚拟试衣技术作为连接线上购物体验与线下真实试穿效果的关键桥梁,近年来受到学术界与工业界的广泛关注 [2]。针对服装电商行业所暴露的线上虚拟试衣问题,曲欣等 [3] 设计的 AR 试衣方案虽能实现三维人体建模与虚拟服饰叠加,但该技术存在设备依赖性强、计算资源消耗大,以及跨平台兼容性不足

等缺陷,难以覆盖移动端多样化场景。林凤秒 [4] 等设计的 VR 试衣技术虽然方便消费者查看试衣效果,但仅能在提前设定好的人物模特上进行虚拟试衣,与消费者本人试衣的效果有较大出入。在跨平台框架方面,Flutter 因 Dart 语言 [5] 生态薄弱导致开发效率低,React Native 的桥接机制可能导致界面渲染延迟 [6]。综上所述,现有研究工作普遍面临两大挑战:技术方案轻量化不足,以及系统架构缺乏跨平台一致性。

针对上述挑战,本文提出了一种基于 UniApp 框架的轻量化跨平台虚拟试衣系统解决方案。UniApp 是基于 Vue.js 的跨平台应用开发框架,支持通过单一代码库编译生成 iOS、Android、Web 及各类小程序应用,有效兼顾了性能与多端兼容性 [7]。后端采用轻量级 Flask 框架 [8],将 OOTDiffusion 模型 [9] 封装为 RESTful API 服务,在充分利

【基金项目】省级大学生创新创业项目(项目编号: No. S202410709076)。

【作者简介】李兵兵(2003-),男,中国甘肃天水人,本科,从事计算机科学与技术研究。

用其原生强大图像生成能力的同时，规避了传统方法中复杂的模型重构过程。本文所实现的系统充分利用 UniApp 的跨端编译能力，实现了 Web、iOS 和 Android 平台的无缝适配，相较于单一平台方案显著提升了用户覆盖率；其次，本系统在严格保持 OOTDiffusion 原始模型生成精度的前提下，通过客户端/服务器 (C/S) 架构与客户端的轻量化设计，显著降低了移动端计算负载，确保系统即使在低端移动设备上也能流畅运行。

2 系统分析

系统的虚拟试衣功能涵盖上衣、下装及连衣裙三大类别。用户可根据需求选择试衣类别，系统将据此生成相应的虚拟试穿效果。图片上传功能需支持用户通过移动端应用或 Web 端页面提交本地图片（包含用户照片与衣物照片），移动端还需支持调用摄像头实时拍摄上传。用户应能实时预览并更换已上传的图片。后端系统在接收到前端传输的用户照片、衣物照片及选定的试衣类别后，调用 OOTDiffusion 模型生成多张贴合用户身材的试衣效果图。前端以轮播图形式展示生成结果，用户可通过左右滑动浏览并选择最优方案。生成的虚拟试衣效果图需支持高清下载及社交媒体分享功能。

为了限制 Android 端峰值内存消耗不超过 150MB，我们采用 Web Worker 线程隔离与模型按需加载机制优化资源使用。此外，为了保证用户个人信息不泄露，对用户上传的原始图像在前端进行脱敏处理（移除 EXIF 地理位置等敏感元数据），数据传输过程采用 HTTPS 协议加密。

3 功能实现

3.1 前端实现

基于 UniApp 框架构建跨平台前端模块，采用 Vue3 组合式 API 实现响应式数据流控制，通过组件化开发模式将功能模块解耦。

3.1.1 用户界面

界面采用分层式布局设计，分层式布局，allBox 容器包含导航区（集成品牌标识与标题，flex 布局适配多端）、主操作区（上传面板与结果展示面板双卡片设计）；上传区域支持 jpg 格式图片上传，上传后显示预览；结果展示区呈现多张试衣效果，用户可切换查看；折叠控制模块通过 v-show 指令实现面板展开/收起切换。

3.1.2 图片上传组件

图片上传功能通过 UniApp 的 uni.chooseImage API 实现基于 Promise 的异步文件处理流程。用户可以分别上传人物图片和衣服图片，选择完成后会展示在界面上，并通过 uni.uploadFile API 上传到服务器。上传过程中显示加载动画，确保用户了解当前处理状态。

在人物照片上传组件中，用户点击“上传人物照片”后，弹出文件选择框，上传成功后显示上传的图片预览。衣物照

片上传组件，与人物照片上传类似，用户点击“上传衣服照片”上传图片。

3.1.3 记录保存

上传的图片会保存在组件的 data 中，personImg 和 clothImg 分别存储人物和衣服的图片路径。每次选择图片后，路径会被更新并展示出来。通过 uni.uploadFile 将图片上传至后端，确保数据得到保存并能进行后续处理。

3.1.4 优化策略

用 CSS3 动画（如等待旋转效果）替代 JS 动画降低开销；v-for 指令循环渲染 radio-item 组件，减少 DOM 节点创建与销毁次数，提升列表渲染性能；组件卸载时（监听 onUnload 生命周期）清理临时文件引用，避免内存泄漏；LRU 策略管理预览图像缓存，自动淘汰久未使用项，平衡加载效率与内存占用。

3.2 服务端实现

系统服务端采用基于 Flask 框架的微服务架构，通过 RESTful API 规范实现前后端解耦。服务端核心功能包含图像接收、模型调度与结果生成三个子系统，采用多进程架构处理并发请求。在 API 设计层面，遵循 Richardson 成熟度模型 Level 2 标准，通过 HTTP 动词精确表达资源操作语义，例如使用 POST 方法创建试衣任务，GET 方法获取生成结果，形成完整的请求 - 处理 - 响应闭环。

图像接收部分设计了两个上传接口，即 /upload_person_image 和 /upload_clothing_image，分别用于接收人物图片和衣服图片。图片通过 POST 请求上传至服务器，服务端使用 request.files 获取上传的图片文件。接收到图片后，服务器会将其保存到本地的临时目录中，并返回上传成功的消息以及图片保存路径。此设计保证了文件的接收和存储过程清晰可控。

服务器接收到用户传来的完整“人物”+“衣物”图片后，通过 /virtual_try_on 接口来启动虚拟试衣过程。该接口会触发模型调用逻辑，通过调用 run_ootd_model() 函数，启动一个外部的 Python 脚本（run_ootd.py），该脚本负责执行虚拟试衣的核心任务。任务完成后，模型生成的试衣效果图会保存到指定的目录中。在当前实现中，该接口并没有引入复杂的任务队列管理方案，所有的图像生成任务都在用户请求时同步执行。

3.3 试衣模型的集成

模型的部署环境方面，采用了支持 GPU 加速的框架以提升推理效率。具体来说，OpenPose 和 Parsing 模型，以及 OOTDiffusionHD 和 OOTDiffusionDC 两种虚拟试衣模型，都可以在 GPU 上运行。通过传递 GPU 的设备 ID（例如 OpenPose(0)），确保模型能够利用显卡的计算能力执行复杂的图像处理任务，提高计算效率。

输入输出规范方面，设计了详细的图像预处理和后处理流程。在预处理阶段，首先对输入的衣物图像和人物图

像进行缩放操作,确保输入图像符合模型的预期输入尺寸(768x1024)。人物图像经过 OpenPose 模型处理后,生成关键点数据,并且通过 Parsing 模型获取分割信息。经过上述操作,生成可以与衣物图像合成的准确人像轮廓,从而为虚拟试衣过程提供更精细的控制。get_mask_location() 函数用于生成遮罩图,用于定义人物区域和背景区域,方便后续的合成工作。最终,这些预处理结果会输入到虚拟试衣模型(如 OOTDiffusionHD 或 OOTDiffusionDC)中进行生成。

模型调用部分,根据用户选择、上传的试衣类别,选择对应的虚拟试衣模型进行推理。在运行时,代码会根据传入的参数配置(如类别、步数等),自动选择适当的模型类型(hd 或 dc),并调用相应的模型。模型根据输入的衣物图像、人物图像和遮罩等信息,生成合成后的试衣效果图。模型执行过程中,图像被分批处理,并且生成的图像保存在指定的输出目录中。

后处理逻辑方面,生成的图像会经过简单的文件保存操作,并且可以进一步优化图像的显示效果。图片会以 PNG 格式保存在 images_output 目录下,之后返还给前端。

3.4 运行效果

本系统所有的测试工作均在普通安卓手机环境下进行。测试设备为一款中低端安卓智能手机,系统版本为 Android 11,以确保系统的普遍适用性和稳定性。测试主要围绕功能性测试和性能测试两个方面展开。

功能测试主要是测试系统的上传图片、结果生成、下载结果及分享功能。软件界面如图 1 所示。人物及衣服图片的上传功能如图 1(a)(b)所示,当用户上传人物图片、T 恤衫图片和裤子图片后,后端服务器成功接收到图片。上传图片并点击生成按钮后,后端服务器调用 OOTDiffusion 模型生成多个虚拟试衣结果,前端以轮播图的形式展示出来,如图 1(c)(d)所示。用户点击下载按钮,即可下载对应的试衣结果;点击分享按钮,即可将结果分享至社交平台。

性能测试则集中在端到端延迟和内存占用两个重要指标上。对于端到端延迟,经过多次测试,生成过程的平均时间控制在 30 秒以内,符合要求。这意味着从用户上传图片到最终生成试衣效果图的全过程能够在 30 秒以内完成,保证了系统的高效性。在内存占用方面,测试结果表明,Android 端的系统内存占用大约为 200MB 左右,属于正常范围内,不会导致手机出现卡顿或过度的资源消耗。

4 结语

本文所设计系统能够根据用户上传的衣物图像和人物图像,利用不同的虚拟试衣模型(如 OOTDiffusionHD 和 OOTDiffusionDC)进行图像合成,生成与真实试穿效果相近的试衣效果图。然而,系统也存在一定的局限性。首先,当前的生成效果主要依赖于现有的模型架构,在处理某些复杂服装类型或特定体型时,仍可能出现不自然的配合或细节

丢失。例如,针对不同体型的服装合成效果,模型的适应性和准确性仍然有所限制。其次,尽管本系统在大多数情况下能够生成较为真实的效果图,但面对实际的多样化试穿场景(如光照变化、服装纹理复杂度等),仍可能产生一些不理想的合成图像。

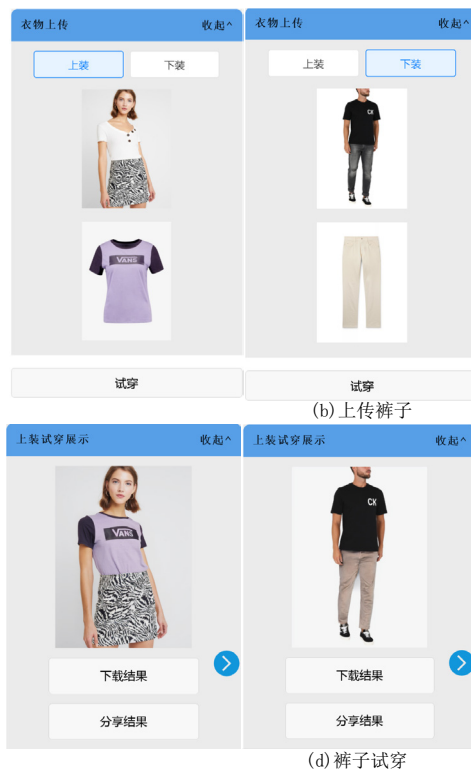


图 1 软件界面

参考文献

- [1] 李照璇.新零售场景营销对消费者行为的影响研究[D].湖南大学,2023.
- [2] 全子言,和羽萱,王晶,等.虚拟试衣在中国服装行业的应用及商业模式创新研究[J].现代商业,2023,(18):15-18.
- [3] 曲欣,刘春艳,齐贝宁,等.AR虚拟试衣技术发展现状及推广前景展望[J].内江科技,2023,44(01):87-88.
- [4] 林凤秒,勇金华,陈艳.基于VR技术的虚拟试衣间设计[J].电视技术,2022,46(09):219-221+228.
- [5] 亢少军.Flutter技术入门与实战[M].机械工业出版社:202006.1257.
- [6] 向治洪.React Native移动开发实战[M].人民邮电出版社:202311.487.
- [7] 滕毅,马焯文.基于Uniapp的校园拼车多端小程序设计与实现[J].电子制作,2022,30(16):43-46.
- [8] 杨洪涛.Flask中ORM模型的应用及研究[J].电脑编程技巧与维护,2023,(10):49-51+75.
- [9] Xu Yuhao, Gu Tao, Chen Weifeng, and Chen Chengcai. OOTDiffusion: Outfitting Fusion based Latent Diffusion for Controllable Virtual Try-on[J]. arXiv preprint arXiv:2403.01779v2, 2024.