

doi:10.3969/j.issn.1673-9833.2020.05.007

基于 ARKit 技术的文物交互式相机 App 设计与实现

张赛东, 邓晓军, 李玉龙, 欧阳子谦

(湖南工业大学 计算机学院, 湖南 株洲 412007)

摘要: 为了让用户在博物馆欣赏文物时更好地了解文物的历史和意义, 采用 iOS 平台最新的 ARKit 增强现实、SwiftUI 全新 UI 框架和 Core ML 机器学习技术, 设计并实现了一款基于 iOS 平台集文物识别与检测、文物 3D 建模和文物交互多功能于一体的 App。系统测试结果表明, 所设计的系统能准确并高效地实现文物识别和检测、文物相关信息的获取、文物 3D 欣赏以及用户与文物之间的互动功能, 为用户带来身临其境的体验, 同时为文物爱好者鉴别和欣赏文物带来了便利, 为实现文物数字化管理提供有力支撑。

关键词: ARKit; SwiftUI; Core ML; 相机; 文物; App

中图分类号: TP311

文献标志码: A

文章编号: 1673-9833(2020)05-0040-07

引文格式: 张赛东, 邓晓军, 李玉龙, 等. 基于 ARKit 技术的文物交互式相机 App 设计与实现 [J]. 湖南工业大学学报, 2020, 34(5): 40-46.

Design and Implementation of Interactive Camera App for Cultural Relics Based on ARKit Technology

ZHANG Saidong, DENG Xiaojun, LI Yulong, OUYANG Ziqian

(College of Computer Science, Hunan University of Technology, Zhuzhou Hunan 412007, China)

Abstract: In order to enable users to better understand the history and significance of cultural relics in the museum, a multifunctional App has thus been designed and implemented based on iOS platform, which integrates cultural relics identification and detection, cultural relics 3D modeling and cultural relics interaction by adopting the latest ARKit, SwiftUI and Core ML technology. The system can accurately and efficiently realize the functions of cultural relic identification and detection, acquisition of cultural relic related information, 3D appreciation of cultural relic and interaction between users and cultural relics. Meanwhile, it can provide users with immersive experience, and facilitate the identification and appreciation of cultural relics for fans of cultural relics, thus providing strong support for the realization of digital management of cultural relics.

Keywords: ARKit; SwiftUI; Core ML; camera; cultural relics; App

收稿日期: 2020-06-17

基金项目: 湖南省自然科学省市联合基金资助项目 (2019JJ60054), 湖南省教育厅普通高校教学改革研究基金资助项目 (湘教通 [2018]436 号-441), 湖南省大学生创新训练基金资助项目 (湘教通 [2019]100 号-2717)

作者简介: 张赛东 (2000-), 男, 湖南耒阳人, 湖南工业大学学生, 主要研究方向为软件与人工智能,

E-mail: 3440217568@qq.com

通信作者: 邓晓军 (1974-), 男, 湖南衡南人, 湖南工业大学教授, 主要从事计算视觉方面的教学与研究,

E-mail: 13873358664@139.com

0 前言

据视+AR网站统计,如今国内从事AR应用开发的企业有200多家^[1]。国内诞生了视辰信息、亮风台、奥图科技等研发和生产AR的软硬件公司,同时也诞生了EasyAR、灵犀AR等致力于某细分领域的技术解决方案的服务商家^[2]。但是在诸多国内AR软硬件公司之中,致力于将AR相机技术与实际的物品进行互动交流的产品较少,致力于将AR相机技术与中国传统文物相结合的产品更是屈指可数^[3]。

目前海量的文物需要在互联网上展示,特殊化、个性化、真实化文物展示便显得尤为重要^[4]。由于如今的3D模型制作成本的制约,这些需求只能暂时以二维照片予以满足,从而造成传递给用户的文物信息不够全面、详实和逼真^[5]。

通过对国内外文献的分析以及市场AR相机类软件的应用,目前基于AR技术类的相机App主要存在以下问题:

- 1) 在传统文化方面的应用较少;
- 2) 与实际的物品基本无交互;
- 3) 绝大多数未使用3D模型展示。

针对上述问题,本研究拟设计一款能与文物进行互动的AR交互式相机App。系统将采用ARKit、Core ML、SwiftUI技术为用户提供友好的交互界面和身临其境般的增强现实体验,相对于目前市场上的AR类App的优势明显。系统专注于文物交互,通过对文物进行三维建模,通过ARKit增强现实技术的方式将文物展示给用户,并且将文物的信息呈现给用户,让用户能够更清楚地了解文物,通过Core ML机器学习技术,识别、检测、渲染真实的文物,让用户能够及时地了解到的文物的相关信息。

1 系统架构

如图1所示,系统架构主要分为3层,从上向下依次为业务层、组件层和SDK层。业务层包含业务模块与业务数据,主要负责对业务模块和业务数据的封装;组件层包含通用工具和领域工具,主要负责对系统组件的封装;SDK层则负责管理系统API。层级之间皆使用接口通信,各层级之间单向依赖,拒绝跨层访问,如此便实现了层与层之间的隔离^[6]。

业务层内各模块之间的通信是基于消息总线和路由总线的,因此实现了模块内隔离。业务层内各模块内部分别定义自己的消息代理、路由代理、实现内部和外部的消息通信处理^[7]。业务层的业务模块主要包括主模块、文物三维模型模块、文物识别与检

测模块、文物交互模块、设置模块和用户中心模块。业务层的业务数据主要包括用户数据、文物数据、文物三维模型数据、机器学习模型数据、App配置数据、分享数据和日志数据。业务层子模块内部职责划分为M(Model数据,Manager业务逻辑)、V(View,展示与视图逻辑处理)、C(Controller,调度M层,加载V层,刷新View层)、A(ApiManager,消息转发与控制)、消息代理(模块间消息交互代理)和路由代理(模块间路由交互代理)^[8]。

APIManager为系统的API管理模块,主要管控App各模块共享的API服务,使用协议便可调用;HookManager主要用于模块的拦截、横切和解耦^[9];模块管理器主要管控App所有模块的加载顺序、加载状态和生命周期;UI管理器封装了App的通用UI组件,可动态配置和拦截横切;消息总线负责App业务层内部模块间的通信;路由总线负责App业务模块层路由跳转^[10]。

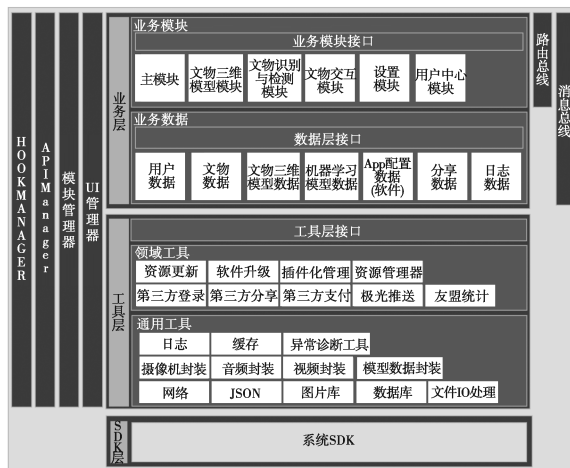


图1 系统架构图

Fig.1 System architecture

2 关键技术

2.1 SwiftUI 全新 UI 框架

SwiftUI 是一种创新、简洁的编程方式,通过Swift的强大功能,在所有Apple平台上构建用户界面。借助它,开发者只需一套工具和API,即可创建面向任何Apple设备的用户界面。系统使用SwiftUI+UIKit完成整个UI界面的实现,由于SwiftUI使用声明式语法,系统的开发变得更为高效和可靠。

2.2 Core ML 机器学习

Core ML可将众多机器学习模型类别整合到开发者的App中。Core ML 3支持高级神经网络,并且能够无缝地使用CPU、GPU和神经网络引擎,提供出

色的性能和效率。除此之外,开发者可以直接在设备上运行机器学习模型。

系统在文物目标检测中使用了目标检测技术,开发者通过爬虫技术获取文物的数据集,而后通过软件或开发者编写的脚本将文物图片中的文物框选出来,获取文物框相对图片的位置,并以文物框左上角的顶点为原点。另外还需获取文物框的长度和宽度属性,便得到了文物在图片中所处的位置和大小信息。利用软件或开发者开发的脚本生成对应的JSON文件,将数据集和JSON文件投入Create ML目标检测机器学习模型训练,具体的训练情况如图2所示。由图可知在目标检测机器学习模型训练过程中,文物目标检测的损失值从8.00降到了0.94,这表明文物目标检测的准确率得到了很大的提高。

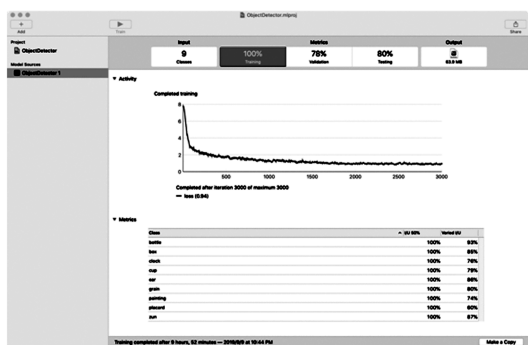


图2 目标检测模型训练图

Fig. 2 Training chart of the object detection model

目标检测的原理和相关算法较多,常见的目标检测方法有光流法、背景差分法和时间差分法,特别的目标检测方法有基于回归的目标检测法、高斯模型提取背景检测目标法、基于霍夫变换和广义霍夫变换的目标检测法、基于霍夫森林的目标检测法、基于ASM(active shape model)的目标检测法、基于Cascade分类器的目标检测法等。目标提取及后续处理的方法主要有图像平滑滤波、阈值选取算法和目标形态学处理算法等,下面简要讲解阈值选取算法中的最大类间方差法。

最大类间方差法的主要原理是计算直方图和归一化直方图,通过直方图的零阶矩 $\omega(k)$ 和一阶矩 $\mu(k)$ 得到最大方差的灰度值,该灰度值即为阈值。对于给定的一幅图像,图像有 M 个灰度值 $[1, 2, \dots, M]$,灰度值为 i 的像素点个数为 n_i ,则有总像素点 $N=n_1+n_2+\dots+n_M$, k 为灰度值门限,灰度值为 $[1, 2, \dots, k]$ 的像素点为背景像素点,用 C_B 表示背景像素点;灰度值为 $[k+1, \dots, M]$ 的像素点为目标像素点,用 C_O 表示目标像素点。故可使用归一化的灰度值直方图,将灰度值为 i 的像素点在图像中出现的概率记为 P_i ,

即视这幅图像的概率分布如式(1)所示:

$$P_i = \frac{n_i}{N}, P_i \geq 0, \sum_{i=1}^M P_i = 1, \quad (1)$$

则出现背景像素点 C_B 的概率 ω_B 的计算式如式(2)所示:

$$\omega_B = \sum_{i=1}^k P_i = \omega(k), \quad (2)$$

出现目标像素点 C_O 的概率 ω_O 的计算式如式(3)所示:

$$\omega_O = \sum_{i=k+1}^M P_i = 1 - \omega(k), \quad (3)$$

其中零阶矩 $\omega(k)$ 的计算式如式(4)所示:

$$\omega(k) = \sum_{i=1}^k P_i. \quad (4)$$

另外背景像素点 C_B 的平均灰度值 μ_A 的计算式(5)所示:

$$\mu_A = \sum_{i=1}^k i P_i / \omega_B = \mu(k) / \omega(k). \quad (5)$$

目标像素点 C_O 的平均灰度值 μ_O 的计算式(6)所示:

$$\mu_O = \sum_{i=k+1}^M i P_i / \omega_O = \frac{\mu - \mu(k)}{1 - \omega(k)}, \quad (6)$$

其中一阶矩 $\mu(k)$ 的计算式如式(7)所示:

$$\mu(k) = \sum_{i=1}^k i P_i. \quad (7)$$

灰度值的均值 μ 的计算式如式(8)所示:

$$\mu = \sum_{i=1}^M i P_i. \quad (8)$$

容易验证,对任意选定的 k ,式(9)皆成立。

$$\omega_B \mu_B + \omega_O \mu_O = \mu. \quad (9)$$

最后有方差 σ^2 的计算式如式(10)所示:

$$\sigma^2 = \frac{[\mu \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]}. \quad (10)$$

最大类间方差法的优点在于算法较简单,当目标与背景的面积相差不大时,能够有效地将图像进行分割。值得注意的是,当图像中的目标与背景的面积相差很大时,直方图没有明显的双峰或双峰相差很大,这便会导致分割效果不佳,其原因在于忽略了图像的空间信息。另外,此方法以图像的灰度为分割图像的依据,对噪声非常敏感,故在实际应用中,一般将此方法同其他方法结合使用。

2.3 ARKit增强现实技术

ARKit 3能够通过People Occlusion功能在人物前后自然地呈现AR内容,该技术最多一次可跟踪3个面孔,并且支持协作会话等功能。现在,开发者可以利用ARKit全新的人物感知能力,将人物动作整

合到 App 中。

系统在文物交互、文物识别与检测、文物 3D 模型中均使用了 ARKit 技术,系统利用 ARKit 为用户提供增强现实技术,为用户带来身临其境的体验感。在文物 3D 模型中,开发者使用 Maya 三维建模技术,建立了文物的三维模型,模型的尺寸、样貌、纹理等都按照现实的文物进行设计。系统使用 ARKit 技术将文物的三维模型展示给用户,ARKit 技术使得用户能够同文物的三维模型进行各种互动,如 360° 欣赏、移动、旋转等。系统利用 ARKit 技术让用户在博物馆之外欣赏文物,为用户欣赏文物带来极大的便利。与此同时,系统还利用 ARKit 技术为用户提供增强现实文字、信息、绘画、渲染等功能,帮助用户同真实的文物进行互动。

在增强现实环境下可使用微积分计算规则三维文物的表面积和体积,以保证虚拟的三维模型尽可能高地还原真实文物。

如图 3a 所示,给出了瓶状文物纵截面的函数图像,依照此图像,不仅能够计算虚拟三维文物的体积,还能够近似计算真实文物的体积。如图 3b 所示,由连续曲线 $y=f(x)$ 、直线 $x=a$ 、 $x=b$ 及 x 轴所围成的曲边梯形绕 x 轴旋转一周后的立体图像即为瓶状的物体,只要 $y=f(x)$ 足够准确,瓶状的物体即可视为与文物具有相同的物理特征。

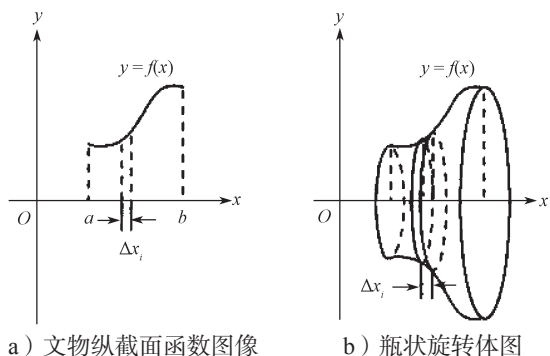


图3 瓶状文物函数图像

Fig. 3 Images of the function of bottle shaped cultural relics

接下来计算文物的体积,取积分变量为 x , $x \in [a, b]$, 在 $[a, b]$ 上任取一小区间 $[x, x+dx]$, 取以 dx 为底的窄边梯形绕 x 轴旋转而成的薄片的体积为体积元素,则可得薄片的体积 dV 的计算式如式(11)所示:

$$dV = \pi f(x)^2 dx, \quad (11)$$

故可以得出旋转体即文物的体积 V 的计算式如式(12)所示:

$$V = \int_a^b \pi f(x)^2 dx. \quad (12)$$

除瓶状文物外,椭球或球状文物也较为常见,诸

如宝石类文物的物理特征都可以用函数表达,下面就椭圆 $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ 所围成的平面图形绕 x 轴旋转一周所成的旋转体(旋转椭球体)的体积进行讨论。这个旋转体可被看成是由半个椭圆 $y = \frac{b}{a} \sqrt{a^2 - x^2}$ 及 x 轴所围成的平面图形绕 x 轴旋转而成的立体结构。则可得椭球状文物的体积 V_1 的计算式如式(13)所示:

$$V_1 = \pi \int_{-a}^a \frac{b^2}{a^2} \sqrt{a^2 - x^2} dx = \frac{4}{3} \pi ab^2, \quad (13)$$

特别当 $a=b$ 时,即文物的形状是球状时,可得球状文物的体积 V_2 的计算式如式(14)所示:

$$V_2 = \pi \int_{-a}^a \frac{a^2}{a^2} \sqrt{a^2 - x^2} dx = \frac{4}{3} \pi a^3. \quad (14)$$

在增强现实环境下需计算三维文物的速度、速率、位移等,以保证用户移动虚拟文物时,文物拥有适宜的速度,画质拥有足够的清晰度。

如图4所示选取的是空间直角坐标系, i, j, k 分别是与 x 轴、 y 轴、 z 轴的正方向同向的单位向量, r 为文物的位置矢量,分别将 x, y, z 记为时间的函数,即 $x=x(t)$; $y=y(t)$; $z=z(t)$, 则有 $r=xi+yj+zk$, 因此文物的运动方程如式(15)所示,该公式可以反映任意 t 时刻文物的位置。

$$r = r(t) = x(t)i + y(t)j + z(t)k. \quad (15)$$

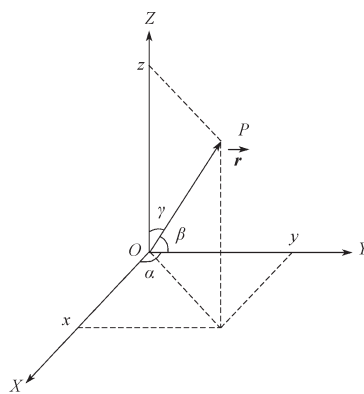


图4 文物位置空间直角坐标系

Fig.4 Spatial rectangular coordinate system for location of cultural relics

在如图5所示的平面直角坐标系中,文物由时刻 t_1 , 起始位置 A 处,经过 $\Delta t=(t_1-t_2)$ 时间后,文物运动到了位置 B 处,文物的位矢由 r_A 变化到 r_B 处。由始点 A 指向终点 B 的有向线段称为点 A 到点 B 的位移矢量,简称位移。可用 r_{AB} 或 Δr 表示。矢量 r_A 、 r_B 、 Δr 刚好为三角形的三边,由三角形法则可以得出文物的位移计算如式(16)所示:

$$\Delta r = r_B - r_A. \quad (16)$$

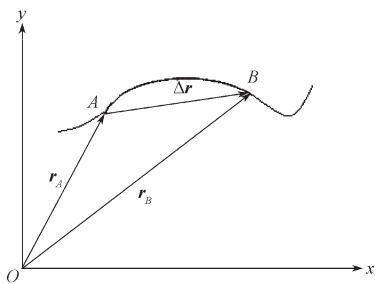


图5 文物位移平面直角坐标系

Fig.5 Plane rectangular coordinate system for displacement of cultural relics

以上述内容为基础,可计算文物的速度,在 Δt 时间内,文物的位移为 $\Delta \mathbf{r}$,则文物在 Δt 时间内的平均速度 \mathbf{v} 如式(17)所示:

$$\mathbf{v} = \frac{\Delta \mathbf{r}}{\Delta t} = \frac{\Delta x}{\Delta t} \mathbf{i} + \frac{\Delta y}{\Delta t} \mathbf{j} + \frac{\Delta z}{\Delta t} \mathbf{k} \quad (17)$$

用户移动文物时,只用平均速度来描述文物的运动显然是不够的,用户若只在某些时刻移动文物过快,靠文物的平均速度并不能发现问题,系统也就不能及时做出处理,故需要知道文物的瞬时速度。

瞬时速度 \mathbf{v}_i 即为平均速度 \mathbf{v} 的极限值,有

$$\mathbf{v}_i = \lim_{\Delta t \rightarrow 0} \frac{\Delta \mathbf{r}}{\Delta t} = \frac{d\mathbf{r}}{dt}, \text{ 即瞬时速度(速度)为位置矢量}$$

对时间的一阶导数,取 $v_x = \frac{dx}{dt}$ 、 $v_y = \frac{dy}{dt}$ 、 $v_z = \frac{dz}{dt}$,则瞬时速度在直角坐标系下的表达式如式(18)所示:

$$\mathbf{v}_i = \frac{d\mathbf{r}}{dt} = \frac{dx}{dt} \mathbf{i} + \frac{dy}{dt} \mathbf{j} + \frac{dz}{dt} \mathbf{k} = v_x \mathbf{i} + v_y \mathbf{j} + v_z \mathbf{k} \quad (18)$$

系统对文物的加速度也颇为重视,将该加速度记为 \mathbf{a} ,则 \mathbf{a} 应该为位置矢量对时间的二阶导数,速度对时间的一阶导数,即 $\mathbf{a} = \frac{d\mathbf{v}}{dt} = \frac{d^2 \mathbf{r}}{dt^2}$,其直系分解形式如式(19)所示:

$$\mathbf{a} = \frac{d\mathbf{v}}{dt} = \frac{dv_x}{dt} \mathbf{i} + \frac{dv_y}{dt} \mathbf{j} + \frac{dv_z}{dt} \mathbf{k} = \frac{d^2 x}{dt^2} \mathbf{i} + \frac{d^2 y}{dt^2} \mathbf{j} + \frac{d^2 z}{dt^2} \mathbf{k} \quad (19)$$

另外,若文物从某定点开始作直线运动,在 t 时间内文物所经过的路程为 $S(t)$,速度为 $v(t) = S'(t)$,其中 $v(t) \geq 0$,则在时间间隔 $[T_1, T_2]$ 内文物所经过的路程 S 可表示为 $S(T_2) - S(T_1)$ 及 $\int_{T_1}^{T_2} v(t) dt$,故可得式(20):

$$\int_{T_1}^{T_2} v(t) dt = S(T_2) - S(T_1) \quad (20)$$

式(20)表明,速度函数 $v(t)$ 在区间 $[T_1, T_2]$ 上的定积分等于 $v(t)$ 的原函数 $S(t)$ 在区间 $[T_1, T_2]$ 上的增量。

在文物的速度、速率或加速度等物理特征不正常

时,系统便会及时进行处理。例如,用户以非常快的速度移动文物时,为防止画质变得模糊,系统便会及时调整文物的速度,使文物的移动过程依旧清晰,这样便能很好地提高用户体验。

3 系统实现及测试

3.1 文物目标检测功能

文物目标检测使用了 ARKit 和 Core ML 来完成对文物方位的检测,使用 Create ML 训练出目标检测模型,利用 Core ML 机器学习使用该模型对屏幕下文物的方位进行检测,并将所检测出的文物的名称、位置、故事和其他相关信息以文字的方式呈现给用户,图 6a、6b 和 6c 分别展示了对青玉云龙纹炉、乾隆款金瓯永固杯和彩漆描金楼阁式自开门群仙祝寿御制钟进行检测的结果界面。



a) 青玉云龙纹炉 b) 金瓯永固杯 c) 祝寿御制钟

图6 文物目标检测界面

Fig.6 Object detection interface of cultural relics

开发人员对文物目标检测功能进行了测试,由于不方便使用真实文物来进行测试,且该功能对文物的图片同样适用,故使用文物的图片代替真实文物来进行测试,测试的数据如表1所示。

表1 文物目标检测功能测试表

Table 1 Function test table for object detection of cultural relics

检测对象	图片像素	检测时间/s	检测结果	置信度
青玉云龙纹炉	240 × 155	1.24	青玉云龙纹炉	0.81
青玉云龙纹炉	480 × 310	1.10	青玉云龙纹炉	0.85
青玉云龙纹炉	960 × 620	0.82	青玉云龙纹炉	0.93
青玉云龙纹炉	1 920 × 1 240	0.36	青玉云龙纹炉	0.98
金瓯永固杯	175 × 202	1.32	金瓯永固杯	0.79
金瓯永固杯	350 × 404	1.12	金瓯永固杯	0.83
金瓯永固杯	700 × 808	0.94	金瓯永固杯	0.88
金瓯永固杯	1 400 × 1 616	0.45	金瓯永固杯	0.95
祝寿御制钟	225 × 205	1.19	祝寿御制钟	0.82
祝寿御制钟	450 × 410	1.06	祝寿御制钟	0.86
祝寿御制钟	900 × 820	0.72	祝寿御制钟	0.94
祝寿御制钟	1 800 × 1 640	0.33	祝寿御制钟	0.98

表1中的数据表明,检测时间与图片像素成反比,置信度与图片像素成正比,即图片越清晰,检测的时

间越短,检测的准确率越高。这里的置信度指的是检测结果同所检测对象的实际情况一致的概率。

3.2 文物三维模型功能

文物三维模型主要使用了 ARKit 技术和三维建模技术,使用 Maya 建立文物的三维模型,利用相关的插件导出 dae 格式,再使用 Xcode 将 dae 格式转换成 scn 格式,投入工程使用。利用 ARKit 扫描出用户周围的平面,待用户选择完所要放置的文物后,使用 ARKit 技术将三维虚拟文物放置到平面之上,并提供各种手势供用户交互使用。实际效果如图 7a、7b 和 7c 所示,展示了王蜀宫妓图、多文物(包括郎窑红釉穿带直口瓶、张成造款雕漆云纹盒和平复帖)和紫檀百宝嵌花果纹长方盒的三维模型。



图 7 文物三维模型界面

Fig. 7 Three-dimensional model interface of cultural relics

开发人员根据真实文物的情况,获取文物近似的纵截面函数图像和高度等信息,通过本文 2.3 节所介绍的规则三维文物的体积计算等相关方法,来计算文物的表面积和体积等能够对文物进行描述的重要信息。凭借这些信息,开发人员便能建立较好的文物三维模型,并且能够对三维模型进行修改和完善,从而保证虚拟的三维模型尽可能高地还原真实文物。

用户使用文物三维模型功能时,在增强现实的环境下,可以进行互动,能够完成移动文物、旋转文物、放大或缩小文物、摆放多个文物等各种交互动作。交互过程中,系统所显示的交互画面越清晰,用户的体验越好。考虑到用户可能会非常快速地完成交互动作,便需要对各种交互动作所产生的效果进行限制,以保证交互过程中画质清晰。

通过 2.3 节所介绍的文物位置、位移、平均速度、瞬时速度、加速度和路程等信息的计算方法,系统可以实时获取上述信息,而当其中某些信息即将出现异常时,系统便会采取相关限制措施,使各信息保持正常。下面举例说明,一般情况下,用户用手指移动文物的速率越快,在增强现实环境下的虚拟文物的移

动速率越快。但是若用户移动虚拟文物的速率过快,便容易出现移动文物时画质不佳、虚拟文物位置信息出错、系统卡顿等问题,这时就需要对虚拟文物的速率进行限制(限制措施可以是设置虚拟文物的速率上限等)以保证系统正常运行,从而确保用户有好的交互体验。

课题组对文物三维模型功能进行了测试,由于手指移动文物的速率不易测量,故使用触控笔代替用户手指进行测试,测试的数据如表 2 所示。

表 2 文物三维模型功能测试表

Table 2 Function test table for three-dimensional model of cultural relics

文物三维模型	模型大小 / MB	触控笔移动文物速率 / (cm·s ⁻¹)	文物移动速率 / (m·s ⁻¹)
王蜀宫妓图	1.2	5.0	0.36
王蜀宫妓图	1.2	10.0	0.72
王蜀宫妓图	1.2	15.0	1.00
王蜀宫妓图	1.2	20.0	1.00
果纹长方盒	1.9	5.0	0.36
果纹长方盒	1.9	10.0	0.72
果纹长方盒	1.9	15.0	1.00
果纹长方盒	1.9	20.0	1.00
穿带直口瓶	3.4	2.5	0.18
穿带直口瓶	3.4	7.5	0.54
穿带直口瓶	3.4	12.5	0.90
穿带直口瓶	3.4	15.0	1.00

表 2 中数据表明,文物三维模型在增强现实环境下移动的速度与模型大小无关。若触控笔的移动速率小于 13.9 cm/s,文物的移动速率与触控笔移动文物的速率成正比,即触控笔移动得越快,文物移动得越快;若触控笔的移动速率大于 13.9 cm/s,文物移动速率会被系统限制在 1.00 m/s,以确保文物的移动画面足够清晰。

4 结语

本研究以帮助人们欣赏文物和对文物进行数字化管理为出发点,针对人们在博物馆或其他地方欣赏文物时对文物缺乏了解的问题,设计并实现了一个基于 iOS 系统的 AR 文物交互式相机 App,系统设计目标明确。在系统编码实现阶段,不仅使用了 iOS 原生开发技术,还运用了目前最新的技术,如在文物识别与检测中使用了神经网络,通过 coremltools 将相关模型转换成 mlmodel,配合 Core ML 使用。最后系统使用 TestFlight 进行测试,检测系统在功能上是否符合预期目标,经过大量的测试以及验证,完成了一定的修正以及优化,目前系统已上架 Apple App Store,为用户提供体验服务。

参考文献:

- [1] 刘立芳, 赵丽莎, 齐小刚, 等. 基于排队模型的网络性能仿真与安全评估技术 [J]. 系统仿真学报, 2018, 30(1): 191-196.
LIU Lifang, ZHAO Lisha, QI Xiaogang, et al. Security Evaluation Techniques and Network Performance Simulation Based on Queue Model[J]. Journal of System Simulation, 2018, 30(1): 191-196.
- [2] 于瑞云, 王鹏飞, 白志宏, 等. 参与式感知: 以人为中心的智能感知与计算 [J]. 计算机研究与发展, 2017, 54(3): 457-473.
YU Ruiyun, WANG Pengfei, BAI Zhihong, et al. Participatory Sensing: People-Centric Smart Sensing and Computing[J]. Journal of Computer Research and Development, 2017, 54(3): 457-473.
- [3] 李 静. 第 41 次《中国互联网络发展状况统计报告》发布 [J]. 中国广播, 2018(3): 96.
LI JING. The 41th Statistical Survey Report on Internet Development in China Released[J]. China Broadcasts, 2018(3): 96.
- [4] 李 霖, 周兴铭. 移动数据库系统的三级复制体系结构 [J]. 计算机学报, 1997, 20(增刊 1): 57-64.
LI Lin, ZHOU Xingming. A Three-Tier Replication Architecture for Mobile Database Systems[J]. Chinese Journal of Computers, 1997, 20(S1): 57-64.
- [5] 承 骁, 白光伟, 华志翔, 等. 云代理的移动消息推送服务 [J]. 小型微型计算机系统, 2016, 37(8): 1661-1666.
CHENG Xiao, BAI Guangwei, HUA Zhixiang, et al. Cloud-Based Broker for Mobile Message Pushing[J]. Journal of Chinese Computer Systems, 2016, 37(8): 1661-1666.
- [6] 王 赞, 郜 健, 陈 翔, 等. 自动程序修复方法研究述评 [J]. 计算机学报, 2018, 41(3): 588-610.
WANG Zan, GAO Jian, CHEN Xiang, et al. Automatic Program Repair Techniques: a Survey[J]. Chinese Journal of Computers, 2018, 41(3): 588-610.
- [7] 孟 清, 杨树强, 蔡建宇, 等. 面向海量数据统计分析查询的中间层语义缓存 [J]. 计算机工程, 2005, 31(20): 67-69.
MENG Qing, YANG Shuqiang, CAI Jianyu, et al. Mid-Tier Semantic Caching for Massive Information Statistic Query[J]. Computer Engineering, 2005, 31(20): 67-69.
- [8] 胡 辉. IOS 环境下使用 MVC 模式进行 APP 开发的设计思路探索 [J]. 数字技术与应用, 2015(6): 204, 206.
HU Hui. Exploration on the Design of App Development Using MVC Mode in IOS Environment[J]. Digital Technology and Application, 2015(6): 204, 206.
- [9] 邢月秀, 胡爱群, 王永剑, 等. 多维度 IOS 隐私泄露评估模型研究 [J]. 网络与信息安全学报, 2016, 2(4): 73-79.
XING Yuexiu, HU Aiqun, WANG Yongjian, et al. Research on Multi-Dimensional IOS Privacy Disclosure Evaluation Model[J]. Chinese Journal of Network and Information Security, 2016, 2(4): 73-79.
- [10] 张李佼. IOS 系统数据的安全性分析 [J]. 中国新通信, 2014, 16(20): 55.
ZHANG Lijiao. Security Analysis of IOS System Data[J]. China New Telecommunications, 2014, 16(20): 55.

(责任编辑: 申 剑)