计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(10):123−131 [doi: 10.15888/j.cnki.csa.009271]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

融合目标检测和空间投影的增强现实方法①

陈　琼, 林兴萍, 舒元昊, 胡青阳

(中电海康集团有限公司, 杭州 311100)
通信作者: 陈　琼, E-mail: 22245087@qq.com

摘　要: 实现对固定目标物的注册跟踪方法中, 目前最常使用预制标识物的方法, 或者需要使用集成深度摄像头等配件的专业 AR 设备, 成本较高. 针对现有方法的缺陷, 提出一种融合目标检测和空间投影算法的简单协作式混合跟踪注册技术, 先通过深度学习算法进行目标检测得到目标物类型, 再利用传感器位姿信息通过空间投影算法确定目标物 ID, 从而提高了虚拟信息叠加在真实场景中的匹配度和准确性. 基于此算法实现了智慧物联基础设施维护的增强现实应用, 并对灯杆、垃圾桶等目标物进行了实验. 实验结果表明, 本方法可以在普通智能手机及 AR 眼镜上运行, 取得了预期效果, 也避免了预制标识物, 降低了对硬件资源的要求.

关键词: 增强现实; 跟踪注册技术; 目标检测; 深度学习; 空间投影

引用格式: 陈琼,林兴萍,舒元昊,胡青阳.融合目标检测和空间投影的增强现实方法.计算机系统应用,2023,32(10):123–131. http://www.c-s-a.org.cn/1003-3254/9271.html

Augmented Reality Method Combining Object Detection and Spatial Projection

CHEN Qiong, LIN Xing-Ping, SHU Yuan-Hao, HU Qing-Yang

(CETHIK Group Co. Ltd., Hangzhou 311100, China)

Abstract: To register and track fixed objects, the common methods are using prefabricated markers, or using professional AR devices with integrated depth cameras and other accessories, whose costs are high. To address the defects of existing methods, a simple cooperative hybrid tracking and registration technology that integrates object detection and spatial projection algorithm is proposed. Firstly, the object type is obtained by the deep learning algorithm for object detection, and then the specific object ID is determined by the spatial projection algorithm using position and posture information obtained from sensors, which improves the matching degree and accuracy of the virtual information superimposed on the real scene. Based on this algorithm, an AR application for smart IoT infrastructure maintenance is realized and experiments are conducted on objects such as light poles and trashcans. The experimental results show that this method can run on ordinary smartphones and AR glasses, achieving the expected results, avoiding the need for prefabricated markers, and reducing the requirement for hardware resources.

Key words: augmented reality; tracking and registration technology; object detection; deep learning; spatial projection

增强现实 (augmented reality, AR) 1992 年由 Cauldell 等[1] 提出, 通过把文字、模型、图像、视频等虚拟信息实时叠加在真实物体上, 实现真实与虚拟场景的无缝叠加. 为实现增强现实, 需要在三维空间位置中对虚拟信息与真实环境进行匹配, 即跟踪注册技术. 跟踪注册的算法性能决定了 AR 的使用效果[2]. 当前, 尤其是在室外环境下, 跟踪注册算法的性能无法很好地满足 AR 系统的要求[3], 特别是考虑到 AR 系统所处

① 基金项目: 国家自然科学基金 (U20B2074); 浙江省重点研发计划 (2021C03032)
收稿时间: 2023-02-10; 修改时间: 2023-04-20, 2023-05-06; 采用时间: 2023-05-17; csa 在线出版时间: 2023-08-21
CNKI 网络首发时间: 2023-08-22

的实际环境和本身的资源往往受到很大限制. 因此, 在实际应用中通常需要混合使用图像识别、定位等多种技术来实现跟踪注册. 本文提出的基于目标检测和空间投影的增强现实方法, 就是希望通过混合两种跟踪注册的方法, 来较好地解决虚拟信息无法和真实场景精确匹配问题.

1 概念和相关工作

1.1 AR 的物理实现

当前视频透视式 (video see-through) 和光学透视式 (optical see-through)[4,5] 是增强现实在物理上的两种主要实现方法. 智能手机等手持设备一般采用视频透视式, AR 眼镜、AR 头盔等常见可穿戴设备较多使用光学透视式. 为验证本文所描述方法的适用性, 后文分别采用了智能手机和 AR 眼镜来进行实验.

1.2 混合注册跟踪注册技术

注册跟踪算法是增强现实的核心技术之一, 也是增强现实的研究重点[6], 它可分为基于传感器及硬件的、基于计算机视觉的、和混合跟踪注册[7–9], 后者是融合多种技术来获取物体位姿的技术. 相对单一技术, 混合跟踪注册提高了鲁棒性和精度, 但也提升了复杂度[10]. 从多传感融合的方式出发, Brooks 等[11] 将其分为互补式、竞争式和协作式.

互补式融合是通过独立的多个传感器分别获得其局部数据后, 在同系统内组合得到全局的系统位姿. 最为常见的是 GPS (global positioning system, 全球定位系统)、北斗等全球定位系统结合其他传感器, 现代 IMU (inertial measurement unit, 惯性测量单元) 通常集成了加速计, 陀螺仪, 磁力仪等多个传感器, 从而能够方便地获得倾角、加速度和磁偏角等姿态信息. 早在 20 世纪 90 年代末, Feiner 等[12] 的户外 AR 导航应用即通过 GPS、电磁、倾角等传感器实现等. 在此基础上利用 SCATT 算法、扩展卡尔曼滤波器 (EKF) 等方法融合计算机视觉, 以实现跟踪注册也是较为常见的做法. 例如闫兴亚等[13] 提出的自适应跟踪注册算法, 再如邓晨等[14] 利用 2 维地图对传感器获得的位姿进行视觉辅助校正, 邹国良等[15] 通过扩展卡尔曼滤波器融合视线方向的传感器注册与基于海天线特征的视觉跟踪注册.

竞争式融合通过融合工具去处理同一空间坐标系中不同的传感器得到的数据不一致的问题. 航迹融合是指使用多传感融合算法对同一测量目标进行跟踪, 但不受传感器单点失效的影响. 如李秋旭[16] 结合有源与无源传感器对目标跟踪, 提高了复杂环境下的抗干扰能力. 集中式融合方法需要每个传感器都有测量信息时才会进行数据融合, 文献 [17] 即采用了该方法.

协作式融合建立不同参考系中传感器参数的内在联系. 其中, 简单协作式融合用一个传感器的姿态对另一传感器的姿态进行补偿. 该方法较为简单且对系统资源要求低, 如王月等[18] 利用点云和视觉特征融合并结合深度图像信息, 提高算法在快速移动时的鲁棒性. 孙启昌等[19] 在手术导航中使用基于光学定位的跟踪注册算法, 解决虚实融合问题. 复杂协作式融合把原始数据融入基于视觉跟踪注册的闭环中, 如文献 [20] 中将惯性方向测量数据嵌入视觉连续帧中, 预测图像特征点.

1.3 融合目标检测和空间投影的混合注册跟踪

为解决增强实现中的虚拟信息和显示环境的精确匹配问题, 本文阐述的方法通过机器视觉的方法对目标进行检测得到目标类型, 即解决是哪种设备的问题, 然后通过获得融合传感器得到的位姿信息来判断是哪个设备, 即解决同一类型设备中的设备 ID (identity document, 身份标志) 问题. 计算机视觉中的识别和追踪是实现 AR 至关重要的技术[21], 本方法的其主要思路是先利用人工智能 (artificial intelligence, AI) 模型识别移动端预览画面中目标物的类型和像素位置, 然后将目标建筑物简化三维模型投影到预览画面所在的屏幕坐标系中, 得到三维模型在预览画面中的像素位置, 将目标物在预览画面中的像素位置和三维模型在预览画面中的像素位置进行一对一比较, 同时比较目标物和三维模型的类型, 得到与目标物匹配的三维模型, 根据匹配到的三维模型得到目标物的识别属性即目标物 ID, 根据这个目标物 ID 即可从服务器得到目标物的动静态信息, 并返回叠加到目标物上.

本文所述方法可以适用于导航、娱乐、工业、设施维护等场景, 其目标物特别适用于各类具有规格型号的智慧物联设备, 如智慧路灯、充电桩等.

2 系统的设计与实现

2.1 增强现实系统的设计

本增强现实方法和系统的前端 (移动端) 和后端 (服务器端) 主要交互流程如图 1 所示.

步骤 1: 构建目标物的简化三维模型, 三维模型存

储到数据库并建立空间索引, 建立地图系统. 如一个智慧灯杆的模型可简化为一个圆柱体, 一个充电桩可以简化为一个立方体等.

步骤 2: 移动端摄像头对准目标物.

步骤 3: 移动端地理位置、姿态数据、摄像头视角、相机预览画面尺寸等自身硬件设备数据. 其中地理位置为 GPS、北斗等定位系统获取的经纬度和高度数据, 姿态数据是移动端在三维空间中旋转的量, 可以是欧拉角、旋转矩阵、四元数或旋转向量. 本系统实现中的姿态数据是根据智能手机的 ROTATION_VECTOR 传感器获取的表示手机旋转的四元数. 摄像头视角根据相机参数计算得到, 摄像头视角为相机视场角 (field of view, FOV), 相机预览画面尺寸包括画面宽和高等.

步骤 4: 移动端将包括自身当前地理位置、检索半径等参数的请求上传到服务器, 请求附近一定空间范围内目标物的三维模型. 其中, 移动端地理位置和检索半径定义了检索数据的空间范围, 空间参考为三维模型经过坐标转换后的参考坐标系.

步骤 5: 服务器从数据库检索到目标物的三维模型. 将三维模型和移动端地理位置转换同一投影坐标系, 该坐标系由移动端的地理位置确定, 可采用地图形变较小的坐标系.

步骤 6: 服务器将三维模型返回给移动端.

步骤 7: 移动端缓存服务器传回的三维模型.

步骤 8: 移动端使用 AI 目标检测模型, 在相机拍摄的预览画面中识别并标注出目标物的类型和像素位置. 虽然 AI 能识别图像中的目标物的类型, 但是无法确定目标物的 ID.

步骤 9: 把指定坐标系中的三维模型投影到相机预览画面所在的屏幕坐标系中. 即通过空间投影的计算, 得到三维模型在相机拍摄的预览画面中的像素位置.

步骤 10: 一对一匹配 AI 识别结果中目标物的像素位置和三维模型在相机拍摄的预览画面中的像素位置, 并比较目标物和三维模型的类型, 得到与预览画面中的目标物匹配的三维模型, 即找到三维模型与 AI 识别目标物的一一对应关系. 由于三维模型中保存有目标物的 ID 等标识性属性, 因此可以通过对应关系知道 AI 识别的目标物详细信息.

步骤 11: 根据匹配结果, 向服务器发送获取目标物详细信息请求, 该请求包括目标物的识别 ID, 详细信息包括目标物静态和动态信息.

步骤 12: 服务器检索目标物详细信息.

步骤 13: 服务器返回检索到目标物详细信息.

步骤 14: 移动端在预览画面上叠加显示信息.

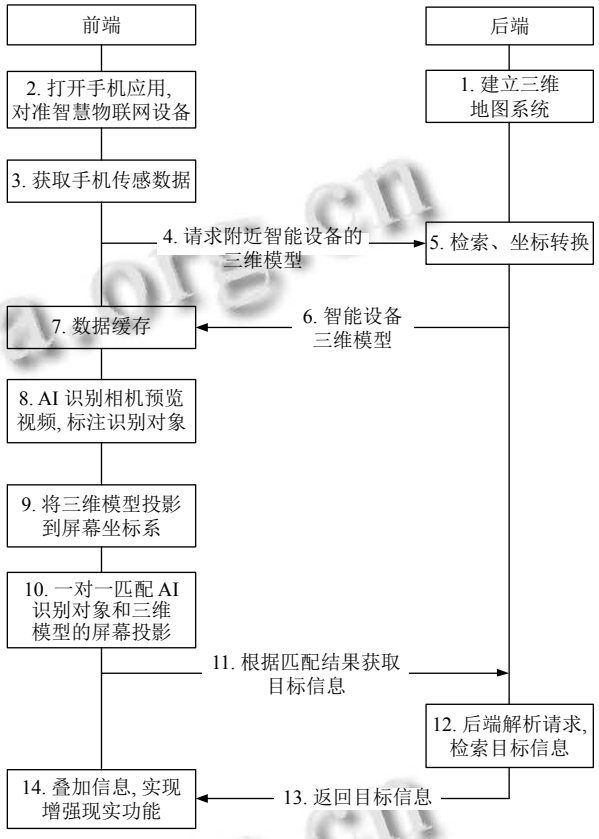

图 1 前后端交互流程

2.2 目标检测的 AI 算法

第 2.1 节步骤 8 的目标检测 AI 算法基于 one-stage 的多尺度特征端到端预测的目标检测算法 SSD[22], 考虑到需要在移动端进行部署, BackBone 选择引入轻量级网络 MobileNet[23], 其优点在于该网络的 1 个 DepthWise 卷积层的每个通道只需和 1 个卷积通道进行卷积操作, 卷积结果再由 PointWise 卷积层做普通的 1×1 卷积, 如式 (1) 所示, W_{DUP}表示该网络的权重参数总量, W_c表示普通卷积的权重参数总量, H, W, C, k, c 分别表示输入特征以及对应卷积核的高, 宽以及通道数, 当 k=3 时, 两者的比值将略小于 1/9. 因此, 通过 DepthWise + PointWise 的拆分, 相比于普通的卷积操作, 该网络的参数量将极大的缩减, 同时也减少了运算量.

$$\frac{W_{DUP}}{W_c}=\frac{H\times W\times C\times\left(k^2+c\right)}{H\times W\times C\times k^2\times c}=\frac{1}{k^2}+\frac{1}{c} \tag{1}$$

其中, H, W, C, k, c 分别代表输入特征的高, 宽通道数以及卷积核的高、宽、通道数目标检测算法的网络结构包含 13 个 DepthWise 卷积层、13 个 PointWise 卷积层以及 9 个普通卷积层, 其中 13 个 DepthWise 卷积层和 13 个 PointWise卷积层交替连接, 第 1 层和最后 8 层均为普通卷积层, 因此一共 35 层卷积层. 所有 DepthWise 卷积层使用 3×3 大小的卷积核, 所有 PointWise 卷积层使用 1×1 大小的卷积核. 每个卷积层的卷积核, 通道、步长等参数设置如表 1 所示.

表 1 网络结构参数

层级	类型	尺寸	步长
1	Conv	3×3×32	2
2	Depth-Conv	3×3×32	1
3	Point-Conv	1×1×32×64	1
4	Depth-Conv	3×3×64	2
5	Point-Conv	1×1×64×128	1
6	Depth-Conv	3×3×128	1
7	Point-Conv	1×1×128×128	1
8	Depth-Conv	3×3×128	2
9	Point-Conv	1×1×128×256	1
10	Depth-Conv	3×3×256	1
11	Point-Conv	1×1×256×256	1
12	Depth-Conv	3×3×256	2
13	Point-Conv	1×1×256×512	1
14	Depth-Conv	3×3×512	1
15	Point-Conv	1×1×512×512	1
16	Depth-Conv	3×3×512	1
17	Point-Conv	1×512×512	1
18	Depth-Conv	3×3×512	1
19	Point-Conv	1×1×512×512	1
20	Depth-Conv	3×3×512	1
21	Point-Conv	1×1×512×512	1
22	Depth-Conv	3×3×512	1
23	Point-Conv	1×1×512×512	1
24	Depth-Conv	3×3×512	1
25	Point-Conv	1×1×512×1024	1
26	Depth-Conv	3×3×1024	1
27	Point-Conv	1×1×1024×1024	1
28	Conv	1×1×512	2
29	Conv	3×3×256	1
30	Conv	1×1×256	2
31	Conv	3×3×128	1
32	Conv	1×1×256	2
33	Conv	3×3×128	1
34	Conv	1×1×128	2
35	Conv	3×3×64	1

目标检测算法 SSD 通过设置不同尺寸比例 (scale) 和高宽比 (aspect ratio) 的锚点 (anchor) 来提取用于边界回归和类别分类的先验边框集合 (default boxes). 由于实际场景中目标物形状多变, 同一目标需要在不同距离下进行检测, 因此需要设计合适的边框尺寸来提高算法的鲁棒性. 对此选择网络结构的第 11, 13, 29, 31, 33, 35 层的输出结果作为特征图 (feature map), 每层特征图提取边框的尺寸比例计算如式 (2), 其中 $S_{\min}=0.1$, $S_{\max}=0.95$, 那么每层对应的尺寸比例为 0.1, 0.27, 0.44, 0.61, 0.78, 0.95. 同时, 为每个边框设计了 9 种不同高宽比, 如式 (3), 比例a_r分别为 1.0、2.0、0.5、3.0、0.333 3、4.0、0.25、5.0、0.2. 由式 (4) 和式 (5) 可以计算边框的 9 种不同高宽, 其高宽比基本覆盖实际场景中数据呈现的形状.

$$S_k = S_{\min} + \frac{S_{\max}-S_{\min}}{m-1}\times(k-1)|_{k\in[1,\delta]} \tag{2}$$

$$a_r = \{1,2,0.5,3,0.33,4,0.25,5,0.2\} \tag{3}$$

$$w = S_k\sqrt{a_r} \tag{4}$$

$$h = \frac{S_k}{\sqrt{a_r}} \tag{5}$$

其中, w 表示边框的宽, h 表示边框的高.

2.3 三维投影的计算方法

前述第 2.1 节步骤 9 中将三维模型投影到屏幕坐标系的步骤如图 2 所示, 各步骤如下.

步骤 1: 移动设备周边指定空间范围内的三维模型集合为集合 D, D 的总数为 N, Dn 为第 n 个元素. 遍历集合 D, 从 n=0 开始, D 中元素和移动设备地理位置的空间参考都为 R. R 在本例为 WGS 84/Pseudo-Mercator 标准. 本方法不限制此处的空间参考必须为 WGS 84/Pseudo-Mercator, 而是需要根据移动设备的实际地理位置选择合适的投影坐标系, 并且在计算中使用移动端获取的姿态数据、摄像头视角、相机预览画面尺寸等数据.

步骤 2: 如 $n<N$ 执行步骤 3, 否则执行步骤 11.

步骤 3: 计算模型变换矩阵 Mm, 用于将三维模型 Dn 转换到世界坐标系. 在本例中, 三维模型是 WGS 84/Pseudo-Mercator 坐标系下的模型. 使用 GPS 采集的相机地理位置是 WGS 84 坐标系下的经纬度, 需要提前转换为 WGS 84/Pseudo-Mercator 坐标系下的米制坐标. 因此三维模型和相机位置的空间参考都为 WGS 84/Pseudo-Mercator, 模型变换矩阵 Mm 为单位矩阵.

步骤 4: 计算视变换矩阵 Mv, 用于将世界坐标系中的三维模型 Dn 转换到相机坐标系. 本例中就是从 WGS 84/Pseudo-Mercator 坐标转换到相机坐标系. 本例使用智能手机函数接口中的方法 *setLookAtM*(*float*[]

rm, *int rmOffset*, *float eyeX*, *float eyeY*, *float eyeZ*, *float centerX*, *float centerY*, *float centerZ*, *float upX*, *float upY*, *float upZ*) 来计算视变换矩阵. 其中视变换矩阵保存在 *rm* 数组中, *rmOffset* 定义了 *rm* 中视变换矩阵的第 1 个值的索引.

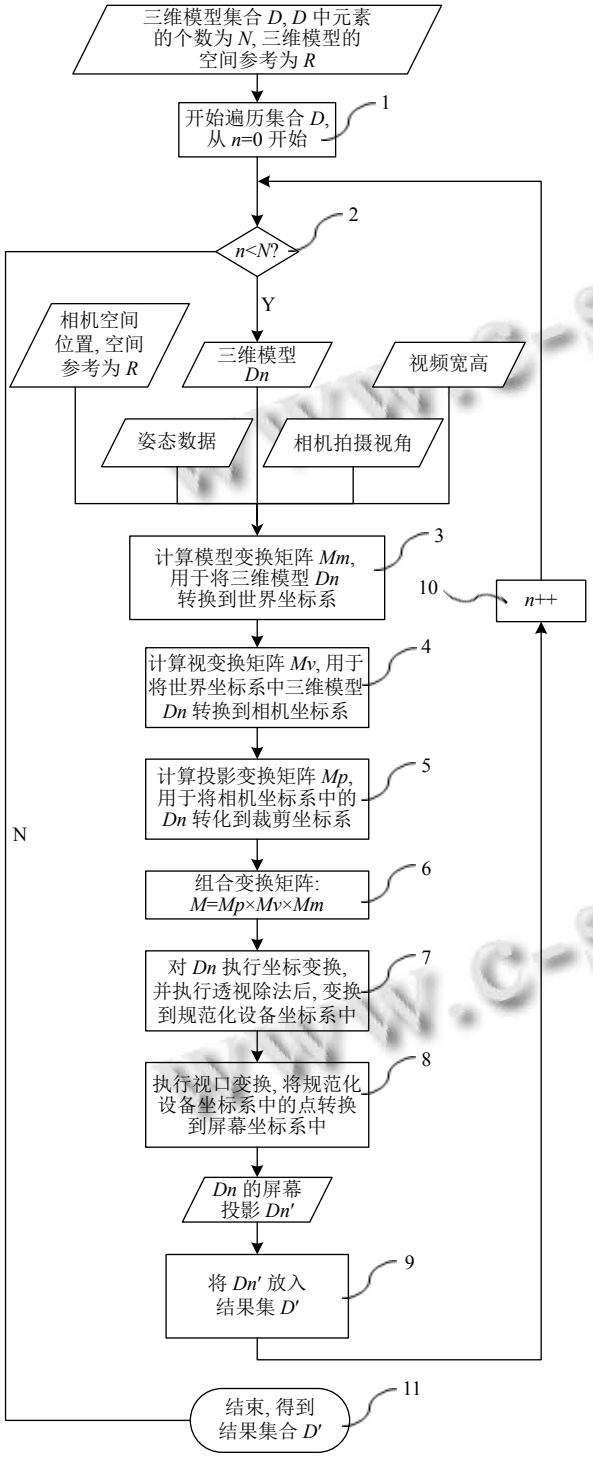

图 2 图三维投影到屏幕坐标系计算流程

如图 3(a) 所示, (*eyeX*,*eyeY*,*eyeZ*)为相机的地理位置坐标, 根据移动端 GPS 获取的 WGS 84 坐标转换到统一的投影坐标系得到, 在本例中是移动端在 WGS 84/Pseudo-Mercator 坐标系中坐标. (*centerX*,*centerY*,*centerZ*)为相机视锥中心的坐标. 如图 3(b) 所示, (*upX*,*upY*,*upZ*)为相机头部朝向是一个向量(*eyeX*,*eyeY*,*eyeZ*)、(*centerX*,*centerY*,*centerZ*)和(*upX*,*upY*,*upZ*)世界坐标系下的坐标. 在本例的以下计算都以 WGS 84/Pseudo-Mercato 为参考系, 则(*centerX*,*centerY*,*centerZ*)算方式如下:

$$p' = (w_1, xi, yj, zk) = qpq^{-1} \tag{6}$$

$$\begin{pmatrix} centerX \\ centerY \\ centerZ \end{pmatrix} = \begin{pmatrix} eyeX + x \\ eyeY + y \\ eyeZ + z \end{pmatrix} \tag{7}$$

式 (6) 为四元数运算, i、j、k 是四元数的虚部, x、y、z 是各虚部的系数, w_1 为实数. q 是手机姿态的四元数表示, 根据智能手机的 ROTATION_VECTOR 传感器获取. q^{-1} 为 q 的逆. $p = (0,0,0,-k)$是三维向量$(0,0,-1)$的四元数表示. (*upX*,*upY*,*upZ*)的计算方式如下:

$$h' = (w_2, upX \times i, upY \times j, upZ \times k) = qhq^{-1} \tag{8}$$

式 (8) 中w_2为实数, $h = (0,0,j,0)$是三维向量$(0,1,0)$的四元数表示.

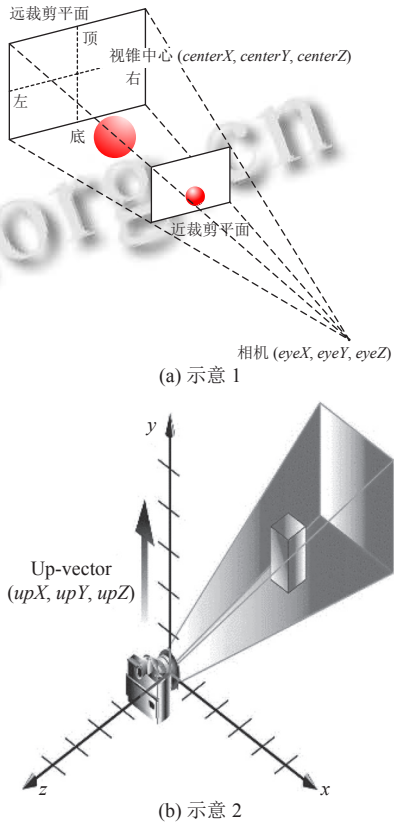

图 3 相机透视成像示意

步骤 5: 计算投影变换矩阵 *Mp*, 用于将相机坐标系中的 *Dn* 转换到裁剪坐标系. 本例中使用智能手机接口中的方法 *perspectiveM* (*float*[] *m*, *int offset*, *float fovy*, *float aspect*, *float zNear*, *float zFar*) 计算 *Mp*. 其中 *m* 是保存 *Mp* 的数组. *offset* 为 *m* 中 *Mp* 的第 1 个值的索引. *fovy* 是相机 *y* 轴方向视角, 如图 3(a) 所示, 底和顶是宽, 左和右是高. 如图 3(b) 所示, Up-vector 的方向就是相机 *y* 轴的方向. *aspect* 为视口的宽高比, *zNear* 为近裁剪平面离相机的距离, *zFar* 为远裁剪平面离相机距离.

步骤 6: 组合各个变换矩阵, 得到综合的坐标变换矩阵 *M*, 其中$M = Mp \times Mv \times Mm$.

步骤 7: 对 *Dn* 执行坐标变换. 三维模型 *Dn* 上的一点为$P = (x, y, z, w)$, *P* 是齐次坐标, *P* 点转换后的坐标为$P' = (Xclip, Yclip, Zclip, Wclip) = M \times P$. 再经过透视除法后, 变换到规范化设备坐标系中, 得到点$Pndc = (Xndc, Yndc, Zndc) = \left(\frac{Xclip}{Wclip}, \frac{Yclip}{Wclip}, \frac{Zclip}{Wclip}\right)$.

步骤 8: 对变换后的三维模型 *Dn* 进行变换, 从规范化设备坐标系转换到屏幕坐标系, 得到投影后的模型 *Dn'*. 也就是将步骤 7 中得到的点 *Pndc* 转换为屏幕坐标$Pscreen = (Xs, Ys)$. 在本例中屏幕坐标系的原点在相机拍摄画面的左上角, 向右为 X 轴正方向, 向下为 Y 轴正方向, 那么 *Pscreen* 的计算公式如下:

$$Xs = screen_w \times \frac{1 + Xndc}{2} \tag{9}$$

$$Ys = screen_h - screen_h \times \frac{1 + Yndc}{2} \tag{10}$$

其中, $screen_w$和$screen_h$分别为相机预览画面的宽和高. 逐一取 *Dn* 上的点, 完成 *Dn* 的坐标转换, 得到转至预览画面所在的屏幕坐标系的三维模型 *Dn'*.

步骤 9: 将 *Dn'*放入结果集 *D'*.

步骤 10: $n = n + 1$, 返回步骤 2.

步骤 11: 程序结束, 得到的集合 *D'*即为三维模型集合 *D* 投影到屏幕坐标系后的集合.

3 智慧物联基础设施应用中的实践

本方法可以被广泛应用在城市基础设施建设运维、工业设备检测、文化商旅等很多领域. 以本文描述的方法为基础, 结合项际需求, 开发了用于智慧物联基础设施的 AR 应用系进行验证. 在本 AR 应用中, 用户需要对智慧物联基础设施设备进行检测和维护. 物联设备包括智慧路灯, 智慧井盖, 智慧垃圾桶等. 如前文所述, 需要对这些智慧物联设备建立简化三维模型. 例如智慧路灯是一个圆柱体, 垃圾桶是一个立方体等.

本 AR 应用实验中, 采用了两种移动终端设备, 分别采用光学透视和视频透视方法. 其中移动终端设备 1 是一款普通的华为 P20Pro 型号的手机, 其配有 6.1 英寸的屏幕和主摄 4 000 万像素的三摄摄像头, 搭载麒麟 970 处理器和 6 GB 内存. 移动终端设备 2 是一个 AR 眼镜工程样机, 如图 4(a) 所示. 该工程样机采用分体式设计, 即其计算单元和电源不放在眼镜上, 而放在与其通过 USB-C 线连接的移动设备上, 该移动设备可以是普通智能手机, 在实验中使用与前相同的华为 P20Pro, 如图 4(b) 所示. AR 眼镜的显示装置为两片采用阵列光波导的 AR 显示模组, 在眼镜上安装摄像头和感知器件, 使得 AR 眼镜能够获得其位置、方向角等信息, 并可传回移动设备进行计算.

(a) 工程样机本体 (b) 工程样机链接智能手机

图 4 AR 眼镜工程样机

3.1 目标检测的 AI 算法训练和结果

在本 AR 应用的目标检测算法中, 一共使用了 10 723 张包含智慧灯杆、垃圾桶或井盖的图片, 将其中的 9 520 张图片作为训练样本集, 1 203 张图片作为测试样本集. 训练过程中设置 BatchSize 为 24, 初始学习率设置为 0.001, 选择 RMS Prop 梯度优化方式, 每 5 万步做一次学习率衰减, 衰减速率为 0.1, 迭代训练 20 万次得到最终的 AI 检测模型. 对测试样本集进行检测, 使用 COCO API (application program interface, 应用程序接口) 进行评估, 评估结果如图 5 所示. 其中整个测试集的 mAP (mean average precision, 平均精准率) 为 0.849, mAR (mean average recall, 平均召回率) 为 0.879, 大物体 (图像像素面积>962) 的 mAP 已经达到 0.866, mAR 达到 0.895; 中等尺寸 (图像像素面积<962 且图像像素面积>322) 的 mAP 为 0.683, mAR 为 0.729; 但小物体 (图像像素面积<322) 的 mAP 和 mAR 却只有 0.467 和 0.5, 可见该模型对小尺寸的物体的准确率仍有待提升. 对智慧灯杆、垃圾桶和井盖识别准确率分别做了评估, 评

估结果如图 6 所示. 其中井盖准确率最高, 达到 0.900 5, 垃圾桶其次, 达到 0.884 7, 智慧灯杆准确率最低, 只有 0.762 8. 相较于垃圾桶和井盖, 智慧灯杆的外形不规则, 呈现细长状, 且从侧面观察时, 如图 7 所示, 方框中背景信息占据大部分特征, 因此引入了更多的噪声.

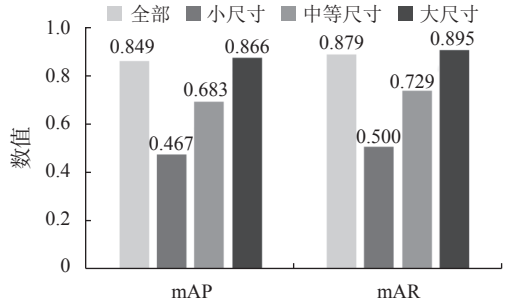

图 5 目标检测的评估结果

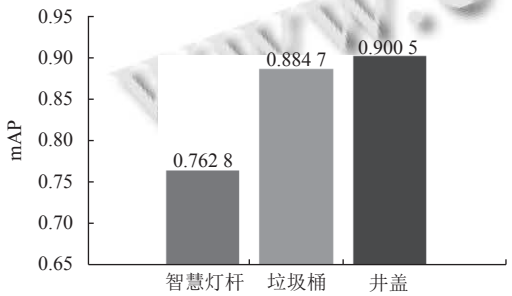

图 6 3 种样本的评估结果

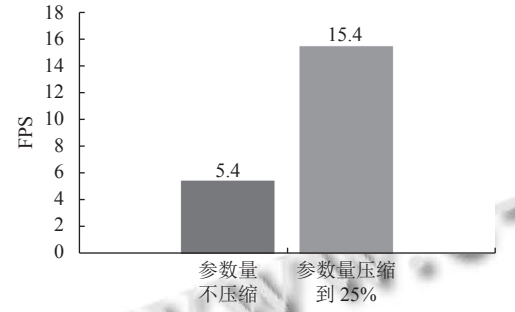

图 7 算法运行速度评估

在实验中, 使用前述智能手机进行目标检测 AI 算法的性能测试, 如果不压缩网络参数量, 每帧处理耗时将达到 200 ms, 无法达到实时效果. 经过测试, 当参数量压缩到 1/4 左右时, 耗时将减少到 70 ms 左右, 此时准确率依然可以被保障.

目前, 算法依然存在不足之处, 例如: (1) 无法识别超过 4 m 以外的井盖和 8 m 以外的垃圾桶, 因此需要提升算法对小尺寸物体的识别能力; (2) 由于老旧井盖常年暴晒雨林, 锈迹较多, 纹路不清晰, 且井盖基本铺设在沥青路中, 因此较难分辨. 算法对这类老旧井盖识别能力相对偏低; (3) 对长条状的智慧灯杆识别能力相对偏低; (4) 目前算法的运行速度仍有待提升.

未来将考虑从以下几方面提升软件算法性能: (1) 扩大老旧井盖、小尺寸等难例样本的数量; (2) 在数据预处理的时候, 使用更多的数据增强方法, 如图像锐化, 增加高斯滤波, 图像去模糊等; (3) 考虑尝试寻找一些更优秀的网络结构替换当前网络, 以求达到更快、更好的检测效果.

3.2 应用的修正和效果

经过 AI 算法对智慧路灯、智慧垃圾桶等目标物的训练及前述第 2.2 节三维投影计算后, 需要一对一匹配 AI 目标检测对象和三维模型的屏幕投影, 在本实验中就是找到目标检测到的智能设备和设备的三维模型之间的一一对应关系. 由于目标物为具有一定体积的物体, 故其在预览画面中的像素位置为具有一定范围的区域, 在理想状态下, 识别结果中目标物的像素位置和匹配的三维模型在相机拍摄的预览画面中的像素位置将会重叠, 即两区域将重叠, 从而得到目标物匹配的三维模型. 但在实际应用中, 识别对象和三维模型投影的屏幕坐标存在偏差, 如图 8 所示. 发生偏差的主要原因是传感器获得的感知信息不太精确, 特别是定位信息不精确, 而方位角、俯仰角和翻转角的精确性也有影响. 由于在 AR 眼镜工程样机中采购的定位和感知器件精确性较手机为好, 所以出现的偏差较低一些.

本应用系统为了降低这种偏差的干扰, 采用修正措施, 即目标物与三维模型的像素位置存在交集, 并且类型相同的方式判定两者为匹配, 两者像素位置不存在交集或类型不同则两者不匹配.

经过目标物与三维模型匹配获得其标志 ID 后, 可以从系统后端取得目标物的基本信息. 如图 9 所示, 图 9(a) 是在智能手机应用中显示一个智慧路灯的相关信息, 图 9(b) 是在 AR 眼镜应用中显示的一个智慧路灯相关信息. 若想进一步取得目标物动静态信息, 可通过交互操作 (智能手机通过荧幕点击, AR 眼镜通过其附带摄像头进行手势识别的方法) 得到.

3.3 与其他方法的比较

在实现对智慧路灯、智慧垃圾桶等具有标准外形的固定目标物进行注册跟踪的方法中, 常见的做法是使用人工标识进行识别, 如文献 [24] 所示的方法, 在实践中, 也采用了类似文献 [24] 的方法进行了对比实验.

使用文字识别技术, 对如智慧路灯等目标物上的标识码进行识别, 如图 10 所示. 由于标识码文字简单, 无论使用 AR 眼镜工程样机, 还是使用智能手机都能够很快精确地识别目标物 ID, 并通过目标物 ID 从系统后端获得其相关信息, 这个方法明显的缺陷是需要事先在所有目标物上贴上标识码. 在实际使用过程中, 必须距离目标物非常近 (通常需小于 0.5 m), 且需寻找到目标物贴着标识码的一面, 将镜头直接对着标识码才能识别. 这种局限很大程度上限制了其使用范围, 并且大幅降低了用户体验.

图 8 目标检测目标物与三维模型投影匹配示意

(a) 智能手机 (b) AR 眼镜

图 9 应用效果

在文献 [25] 中, 作者描述的方法避免使用人工标识, 而使用了基于迁移学习的神经网络算法对目标物进行分类检测识别, 然后结合其使用的 Hololens 设备上的眼动追踪识别和空间映射功能, 确定目标物的位置. 使用这种方法, 需要设备配有深度摄像头和环境感知摄像头的专门 AR 设备, 成本较高. 目前市面上 Hololens 设备的价格超过 3 万元, 昂贵的成本很大程度上限制了其实用性.

(a) 远景 (b) 近景

图 10 带标识码的目标物

本方法使用目标检测与空间投影融合的算法实现目标物的跟踪识别, 既无需预先配置标识, 也无需深度摄像头等特殊设备, 因此, 可以在普通智能手机上使用. 表 2 对几种方法进行了比较.

表 2 几种跟踪注册方法的对比

影响因素	基于标志物的方法 (文献[24])	无标志物的方法 (文献[25])	融合目标检测和空间投影的方法
算力要求	低	高	中
预置标识	是	否	否
分辨同型号目标物不同个体	是	否	是
普通智能手机支持	是	否	是
先验依赖 (模型训练)	否	是	是
识别时需贴近目标物	是	否	否

4 结束语

本文设计和实现了一种融合目标检测和空间投影的增强现实方法, 基于此方法实现了一个用于城市智慧物联基础设施维护的增强现实应用, 并通过使用普通智能手机和 AR 眼镜工程样机两种移动终端进行了

实验验证. 本方法融合了人工智能目标检测和地理信息的空间投影算法, 是一种简单协作式传感器融合的混合跟踪定位技术. 该方法对硬件资源环境要求不高, 适用于有统一外形规格的目标物, 如智慧路灯、智慧垃圾桶, 充电桩等, 在智慧城市基础设施维护、工业设备检测, 商业旅游等领域有较为广泛的应用前景.

参考文献

1 Caudell TP, Mizell DW. Augmented reality: An application of heads-up display technology to manual manufacturing processes. Proceedings of the 25th Hawaii International Conference on System Sciences. Kauai: IEEE, 1992. 659–669.

2 韩玉仁, 李铁军, 杨冬. 增强现实中三维跟踪注册技术概述. 计算机工程与应用, 2019, 55(21): 25–34. [doi: 10.3778/j.issn.1002-8331.1907-0283]

3 Azuma RT, Hoff BR, Neely HE, *et al*. Making augmented reality work outdoors requires hybrid tracking. Proceedings of th 1999 International Workshop on Augmented Reality: Placing Artificial Objects in Real Scenes. Bellevue: A. K. Peters, Ltd., 1999. 219–224.

4 Billinghurst M, Clark A, Lee G. A survey of augmented reality. Foundations and Trends in Human-computer Interaction, 2015, 8(2–3): 73–272.

5 Zhou F, Duh HBL, Billinghurst M. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. Cambridge: IEEE, 2008. 193–202.

6 王涌天, 刘越, 胡晓明. 户外增强现实系统关键技术及其应用的研究. 系统仿真学报, 2003, 15(3): 329–333, 337. [doi: 10.3969/j.issn.1004-731X.2003.03.008]

7 Mahmud N, Cohen J, Tsourides K, *et al*. Computer vision and augmented reality in gastrointestinal endoscopy. Gastroenterology Report, 2015, 3(3): 179–184. [doi: 10.1093/gastro/gov027]

8 Tao C, Zhao MY, Shi QF, *et al*. Novel augmented reality interface using a self-powered triboelectric based virtual reality 3D-control sensor. Nano Energy, 2018, 51: 162–172. [doi: 10.1016/j.nanoen.2018.06.022]

9 Maidi M, Ababsa F, Mallem M, *et al*. Hybrid tracking system for robust fiducials registration in augmented reality. Signal, Image and Video Processing, 2015, 9(4): 831–849. [doi: 10.1007/s11760-013-0508-4]

10 罗斌, 王涌天, 沈浩, 等. 增强现实混合跟踪技术综述. 自动化学报, 2013, 39(8): 1185–1201.

11 Brooks RR, Iyengar SS. Real-time distributed sensor fusion for time-critical sensor readings. Optical Engineering, 1997, 36(3): 767–779. [doi: 10.1117/1.601274]

12 Feiner S, MacIntyre B, Hollerer T, *et al*. A touring machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment. Proceedings of the 1st International Symposium on Wearable Computers. Cambridge: IEEE, 1997. 74–81.

13 闫兴亚, 马柯, 崔晓云. 基于增强现实的自适应跟踪注册方法. 计算机工程与设计, 2021, 42(3): 684–689. [doi: 10.16208/j.issn1000-7024.2021.03.013]

14 邓晨, 游雄, 张威巍, 等. 基于 2D 地图的城市户外 ARGIS 视觉辅助地理配准技术. 测绘学报, 2019, 48(10): 1305–1319. [doi: 10.11947/j.AGCS.2019.20190007]

15 邹国良, 屠正飞, 郑宗生. 基于混合注册方式的海洋环境增强现实系统. 计算机应用与软件, 2016, 33(10): 158–161. [doi: 10.3969/j.issn.1000-386x.2016.10.035]

16 李秋旭. 复杂环境下多传感器目标跟踪技术研究 [硕士学位论文]. 西安: 西安电子科技大学, 2018.

17 Yokokohji Y, Sugawara Y, Yoshikawa T. Accurate image overlay on video see-through HMDs using vision and accelerometers. Proceedings of the 2020 IEEE Virtual Reality. New Brunswick: IEEE, 2000. 247–254.

18 王月, 张树生, 白晓亮. 点云和视觉特征融合的增强现实装配系统三维跟踪注册方法. 西北工业大学学报, 2019, 37(1): 143–151. [doi: 10.3969/j.issn.1000-2758.2019.01.021]

19 孙启昌, 麦永锋, 陈晓军. 基于增强现实的手术导航系统快速标定算法. 计算机应用, 2021, 41(3): 833–838. [doi: 10.11772/j.issn.1001-9081.2020060776]

20 Aron M, Simon G, Berger MO. Use of inertial sensors to support video tracking. Computer Animation and Virtual Worlds, 2007, 18(1): 57–68. [doi: 10.1002/cav.161]

21 高翔, 安辉, 陈为, 等. 移动增强现实可视化综述. 计算机辅助设计与图形学学报, 2018, 30(1): 1–8.

22 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot MultiBox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37.

23 Howard AG, Zhu ML, Chen B, *et al*. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861, 2017.

24 夏德芳, 刘传才. 基于人工标识的移动增强现实配准方法. 现代电子技术, 2015, 38(8): 26–30. [doi: 10.16652/j.issn.1004-373x.2015.08.035]

25 张乐, 张元, 韩燮, 等. 一种免注册标识的增强现实方法. 科学技术与工程, 2020, 20(8): 3149–3156.

(校对责编: 孙君艳)