软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2016,27(11):2929−2945 [doi: 10.13328/j.cnki.jos.004865]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于自然场景在线学习的跟踪注册技术*

桂振文[1], 刘 越[2,3], 陈 靖[2,3], 王涌天[1,2,3]

[1](中国电子科技集团公司 第七研究所,广东 广州 510310)

[2](北京理工大学 光电学院,北京 100081)

[3](北京市混合现实与新型显示工程技术研究中心(北京理工大学),北京 100081)

通讯作者: 桂振文, E-mail: quizhenwen1983@bit.edu.cn, http://www.bit.edu.cn

摘 要: 三维注册是移动增强现实的关键技术之一,提出了一种在线学习的跟踪注册方法,能够精确地对自然场景进行跟踪注册.该方法首先改进 SURF(speeded up robust features)描述符匹配方法,提高初始注册矩阵的正确性;然后,通过对场景进行有效的在线学习,提高注册精度;最后,利用前一帧的注册矩阵快速恢复已丢失的关键点,以提高注册的速度.实验结果表明,该方法能够较为流畅地对视频帧进行跟踪,并能保持较好的注册精度.

关键词: 跟踪注册;SURF(speeded up robust features)描述符;在线学习

中图法分类号: TP391

中文引用格式: 桂振文,刘越,陈靖,王涌天.基于自然场景在线学习的跟踪注册技术.软件学报,2016,27(11):2929−2945. http://www.jos.org.cn/1000-9825/4865.htm

英文引用格式: Gui ZW, Liu Y, Chen J, Wang YT. Online learning of tracking and registration based on natural scenes. Ruan Jian Xue Bao/Journal of Software, 2016,27(11):2929−2945 (in Chinese). http://www.jos.org.cn/1000-9825/4865.htm

Online Learning of Tracking and Registration Based on Natural Scenes

GUI Zhen-Wen[1], LIU Yue[2,3], CHEN Jing[2,3], WANG Yong-Tian[1,2,3]

[1](No.7 Research Institute, China Electronics Technology Group Corporation, Guangzhou 510310, China)

[2](School of Optoelectronics, Beijing Institute of Technology, Beijing 100084, China)

[3](Beijing Engineering Research Center for Mixed Reality and Advanced Display (Beijing Institute of Technology), Beijing 100081, China)

Abstract: Registration is a fundamental technology for augmented reality. In this paper, a registration approach is proposed to accurately track the natural scenes. The matching method of SURF (speeded up robust features) descriptor is first improved to keep the initial registration matrix validity. Then, effective online learning of the scenes is used to improve the registration accuracy. Lastly, the registration matrix of the previous frame is utilized to rapidly restore the lost key points and accelerate the speed of registration. Experimental results show that the proposed method can keep smooth tracking for video frames and maintain high accuracy of registration.

Key words: tracking and registration; SURF (speeded up robust features) descriptor; online learning

增强现实(augmented reality,简称 AR)是指将计算机生成的虚拟场景实时、精确地叠加到用户所观察到的真实世界景象当中,增强和提升人类对外界的感知能力[1].如何实时、精确地计算摄像机相对于真实世界的位置

* 基金项目: 国家自然科学基金(61072096, 60903070); 国家高技术研究发展计划(863)(2013AA013802); 国家“十二五”重点科技攻关项目(2012ZX03002004); 广东省协同创新与平台环境建设专项(2014B090901024)

Foundation item: National Natural Science Foundation of China (61072096, 60903070); National High-Tech R&D Program of China (863) (2013AA013802); Key Science-Technology Project of the National ‘Twelfth Five-Year-Plan’ of China (2012ZX03002004); Collaborative Innovation and Platform Environment Construction Major Project of Guangdong Province (2014B090901024)

收稿时间: 2014-03-04; 修改时间: 2014-06-28; 采用时间: 2014-09-10

与姿态信息,并利用这些信息将虚拟场景放置到它应所处的位置上,即三维注册(虚实配准),是开发增强现实系统的重点和难点.事实上,三维注册已经成为限制增强现实走向更广泛应用的一个关键问题[2].而构造一个成功的增强现实系统的关键技术是准确的三维注册,即,将真实世界场景与计算机生成的虚拟信息进行准确的融合配准.

目前,在增强现实系统的跟踪注册技术研究中,基于特征的注册研究占主要部分[3].基于特征的注册方法是通过在二维图像特征点与三维场景特征点之间建立一一对应关系,求解摄像机注册矩阵.现有基于特征的注册技术可以分为两类:基于特殊标识的注册和基于自然特征的注册.基于特殊标识的注册方法最具代表性的是由日本广岛城市大学与美国华盛顿大学联合开发的增强现实工具包 ARToolKit[4]和由加拿大国家研究院开发的ARTag[5].这类方法需要在真实环境中放置人工标志物,通过对标志物特征的提取获得注册所需的信息从而达到注册目的.然而在真实环境中放置人工标志,利用标志物进行跟踪注册具有鲁棒性差、无法解决遮挡以及环境光照变化所带来的影响等方面的缺点;同时,标识在场景中也带来了视觉污染.另一类基于自然特征的注册方法经常采用基于离线场景模型重构的注册、在线模型重构的注册和基于平面场景的注册.文献[6,7]中,通过离线重构场景的三维模型,在已知真实场景三维模型的基础上,将模型上的 3D 特征点投影到图像平面上,通过优化特征点重投影误差的目标函数,获取摄像机的旋转和平移参数实现场景的跟踪注册.这种算法的局限性在于:目标函数的优化过程有可能陷入局部最小,而无法获得正确的参数估计.文献[8,9]利用 GPU(graphic processing unit)-SIFT(scale-invariant feature transform)对场景重建和光流对特征点进行跟踪,实现了较好的跟踪注册效果.但是,由于其过分依赖专用图形加速器硬件(GPU),同时将特征点匹配和跟踪注册算法分开进行处理,没有充分利用中间计算结果,计算量较大,只能运行在大型的带专用图形处理硬件的服务器上,通用性较差.文献[10,11]使用在线重构注册 SFM(structure from motion)方法,由于其稳定性较差、时间代价高等原因,还没有在增强现实中得到广泛的应用.文献[12,13]基于平面场景的注册将相对摄像机较远的场景(场景本身的深度相对于摄像机的距离可以忽略不计)作为平面来处理,利用图像序列帧间的特征点匹配获取摄像机姿态注册矩阵,实现对场景跟踪注册.但是该方法最主要的缺点是存在误差累积偏移,无法应用于长序列图像的注册跟踪.

在综合分析以上实时算法的基础上,本文提出了一种对自然场景在线学习的实时跟踪注册的算法,充分利用在线场景特点匹配的中间计算结果,对场景进行有效学习,实现场景的三维注册.本文算法通过改进 SURF (speeded up robust features)[14]的特征匹配方法,实现对场景的三维重构和初始注册矩阵的计算;利用对在线场景进行有效学习,实现摄像机跟踪注册矩阵的实时获取;同时,利用相邻图像帧之间的关系,快速恢复光流算法跟丢的图像特征点,确保用来计算注册矩阵的跟踪点数量,提高了注册的精度.

1 跟踪注册算法的框架

自然场景中大多存在有深度的景物,如室外的建筑物、汽车,室内的办公桌、书柜等,因此预先需要对未知场景进行 3D 模型构建和选择少量的关键帧,通过已知场景的三维模型和关键帧信息,能够有效地实现对室外场景的跟踪.本文在构建三维模型的基础上,提出一种对场景进行有效学习的跟踪注册算法,其具体步骤如下:

- 离线阶段

(1) 拍摄真实场景不同视角的 10 幅图像作为关键帧,提取特征点和进行特征描述符匹配.

(2) 重建真实场景的三维结构.

(3) 建立三维模型点 2D-3D 的映射表.

(4) 生成所有场景参考图像的描述符库.

(5) 训练场景 3D 点的权值.

(6) 用所有参考图像描述符训练生成 Flann 决策树.

- 在线阶段

(7) 采集当前场景的图像帧,提取特征点和计算特征描述符.

(8) 用 Flann 决策树进行特征点匹配.

(9) 根据匹配结果,确定该场景是否能被识别.

(10) 如果识别成功,则认为当前场景存在于已经重建好的场景,则进入步骤(12).

(11) 如果识别失败统计识别次数,连续 3 次识别失败,则返回到步骤(1).

(12) 进行在线场景学习,建立场景三维点坐标到当前图像的特征点二维坐标的对应关系.

(13) 采用 P-N-P 算法求取当前图像相对于真实场景的旋转和平移矩阵 R,T.

(14) 将注册矩阵赋值给虚拟相机,进行虚拟与真实场景融合显示.

(15) 利用光流跟踪算法对后续图像帧进行跟踪;若跟踪点数目小于设定阈值,则进行丢失特征点恢复.

(16) 如果后续帧跟踪失败,则进入步骤(7)重新进入提点匹配;否则,进入步骤(12).

算法流程如图 1 所示.

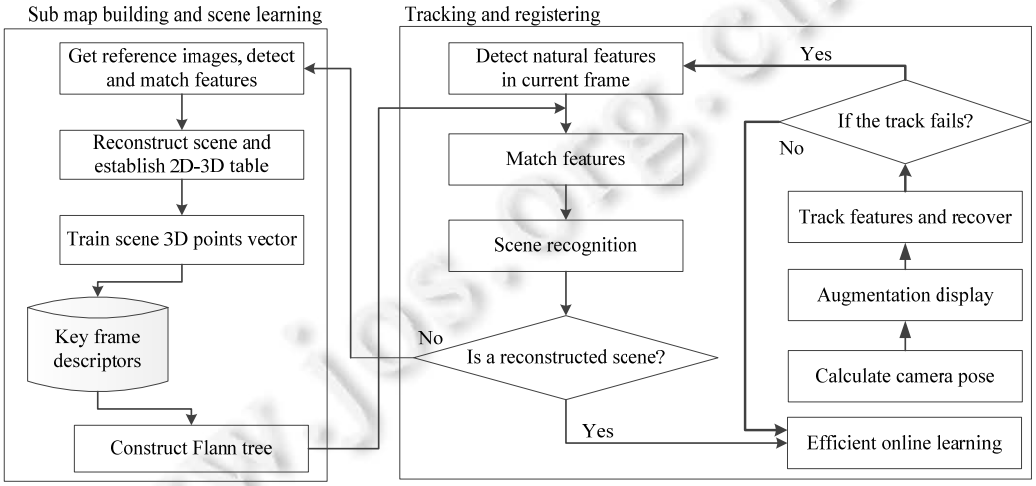


Fig.1 Flowchart of the proposed algorithm

图 1 算法流程图

1.1 离线的场景重建与Flann决策树的创建

离线阶段主要目的是为在线阶段提供场景结构及参考图像描述符等信息,是在线注册过程的前提和基础,该阶段包括:

(1) 场景重建及特征点描述符生成.场景重建过程主要包括选择关键帧、生成场景结构、计算初始位姿矩阵和关键帧的二维特征点位置及对应的三维空间位置).

(2) 建立关键帧特征点的 2D-3D 映射表,方便、快速地查找到 2D 图像特征点对应的空间 3D 位置坐标.

(3) 训练场景 3D 点的初始权值参数.

(4) 训练 Flann(fast library for approximate nearest neighbors)决策树.用所有关键帧的特征点生成 Flann 决策树.在线跟踪时,需要先通过 Flann 决策树对场景进行识别,待场景识别成功后才开始对该场景进行在线学习和跟踪.

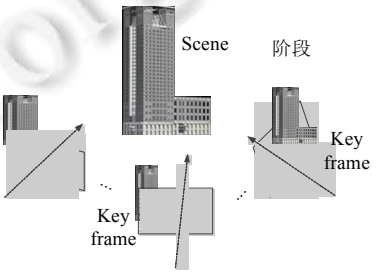


Fig.2 3D scene reconstruction

图 2 3D 场景重建

离线的阶段需要选择一组可以从不同的视角表示应用场景的视图,对这些视图都需要事先进行标定,记录各种重要的已知信息,这样的一组视图就称为关键帧.如图 2 所示的情况,3 个不同视角的视图被选作关键帧.

1.1.1 场景重建

在场景重建前,需要先对摄像机的内部参数矩阵进行标定.文献[15]中,张正友提出的标定方法受限制较少,

加上平面模板的制作简单,应用较为广泛,是 Intel 公司开发的开源计算机视觉库 OPENCV(open source computer vision library)的主要标定方法,在教学和科研中也比较容易实现.本文也是使用该方法,对摄像机的内参进行标定,并且在后续在线跟踪注册中,摄像机内参一直保持不变.

场景的重建具体使用文献[16]中 SFM(structure from motion)的方法,选取关键帧和重建场景的三维结构,但为了使重建场景的三维点和初始的注册矩阵更准确,需要对该方法进行一些修改.在关键帧的选取上,如图 2 所示,选取不同的视点拍摄 10 幅关键帧.在特征提取算法上分析了传统的提点算法,包括 SIFT[17],SURF[14],FAST (features from accelerated segment test)[18],BRIEF(binary robust independent elementary features)[19],ORB(oriented FAST and rotated BRIEF)[20]等等.SIFT 描述符虽然可区分度高,但是由于维度太高,描述符占用的空间比较大,需要消耗较大的内存.这对于场景数和关键帧较多的 AR 应用来讲是非常不利的.FAST 算法复杂度低,可以应用于智能手机平台.但是该算法所提特征点不具有尺度信息,对噪声和尺度变化特别敏感,效率低下.BRIEF 算法在特征描述上采用了与 SIFT 不同的方法,使用二进制串作为特征点描述符,极大地缩短了特征点描述符的计算时间.但 BRIEF 算法对噪声敏感,而且算法不具有方向不变性.ORB 是 BRIEF 的改进算法,给特征点增加了主方向,但是 ORB 特征点并不具有尺度信息.虽然该算法是二进制特征提取算法,生成描述符和匹配阶段速度较快,但该算法的不足之处在于它对方向和噪声比较敏感,不具备旋转不变性,因此对于旋转比较频繁的 AR 的应用而言准确性不高.SURF[14]算法对 SIFT 算法做了一定程度的改进,降低了算法的计算复杂度和描述符的维度,使得在特征点提取与描述符匹配时具有明显的速度优势;同时,该算法对光照变化和透视变换具有部分不变性,也是目前比较流行的宽基线特征点匹配算法之一,并且可以用来解决光流跟踪过程中的积累误差问题.

因此,本文选择 SURF 算法来提取特征点.SURF 算法的特征点描述符用特征向量表示,在进行特征点匹配时,采用传统的欧式距离来计算特征点的相似性,如式(1)所示:

$$d(X,Y)=\sqrt{\sum_{i=1}^{N}(x_i-y_i)^2} \tag{1}$$

其中,$X=(x_1,x_2,\ldots,x_d)$,$Y=(y_1,y_2,\ldots,y_d)$,n 为特征向量的维数(此处,SURF 描述符 n 取 64).

在进行特征点匹配时,通常采用改进的最近邻方法,计算最近邻比与次最近邻的比值来确定两个特征点是否匹配,如果这个比值小于某个阈值,则表示最近距离的两个特征点.该阈值 T_d 通常设置为 0.8 或者 0.6,根据图像内部的结构来决定:图像内部相似纹理结果较多,则设置较低的阈值.具体实现为:设查询图像的所有描述符为 $\{X_1,X_2,\ldots,X_M\}$,场景图像的所有描述符为 $\{Y_1,Y_2,\ldots,Y_N\}$,P 为查询图像中的一点 $X_c(1\leqslant c\leqslant M)$,$A$,$B$ 是场景图像中与 P 点欧式距离最近或次近的两个特征点,两个特征点如公式(2)、公式(3),若 $d(P,A)$满足公式(4),则认为这一对特征点(P,A)为匹配点:

$$d(P,A)=\min\{d(P,Y_1),d(P,Y_2),\ldots,d(P,Y_N)\} \tag{2}$$

$$d(P,B)=\min\{d(P,Y_e)\},\ \text{when}\ Y_e\neq A\ (1\leqslant e\leqslant N) \tag{3}$$

$$\frac{d(P,A)}{d(P,B)}<T_d \tag{4}$$

但是在实际的特征点匹配实验中我们发现,仅用公式(4)来确定匹配点,效果不太好,仍然存在大量误匹配点.我们在国际标准图像库 UKBENCH[21]随机选择同一个场景的两张图像,提取 SURF 特征,按照公式(4)进行匹配,匹配结果如图 3 所示.经过仔细分析发现:对于查询图像特征点 P 来说,特征点 A 是场景图像中与 P 最相似的,但是(P,A)未必是真的匹配点,因为对于特征点 A 来说,特征点 P 未必是目标图像中与特征点 A 最相似的,或许在查询图像中存在点 Q,$d(Q,A)$少于 $d(P,A)$,这样,(P,A)就是误匹配点.所以,本文在场景重建的图像特征点匹配算法中增加如下约束(如下式):

$$d(Q,A)=\min\{d(X_f,A)\},\ \text{when}\ X_f\neq P\ (1\leqslant f\leqslant N) \tag{5}$$

$$\frac{d(P,A)}{d(Q,A)}<T_d \tag{6}$$

Q 点是 A 点与除 P 点以外的目标图像中的所有特征点中的欧式距离最短的点,$d(P,A)$与 $d(Q,A)$满足公式(6),说

明在查询图像中与 A 点最匹配的点也是 P 点.所以,同时满足公式(4)和公式(5)时,无论是相对场景图像还是查询图像,(P,A)都是最佳匹配点,具体效果如图 4 所示.由图 3 和图 4 可以看出,改进的匹配方法剔除了更多的误匹配点,具有更好的匹配精度.

Fig.3 302 matching feature pairs by the original matching algorithm

图 3 最初的特征点匹配算法匹配 302 对特征点

Fig.4 183 matching feature pairs by the improved matching algorithm

图 4 改进的匹配算法匹配 183 对特征点

本文用随机抽样一致性 PROSAC(progressive sample concensus)方法[22]测试了在最初匹配算法和改进后的匹配算法产生的匹配点集上进行几何一致性校验的时间开销.从 UKBENCH 数据集中选取 12 个场景的 24 幅图像,用两种匹配方法进行特征点匹配,对匹配成功的点集用 PROSAC 方法进行几何一致性校验,剔除伪匹配点.PROSAC 方法在 12 个场景的匹配点集上时间开销如图 5 所示,从图 5 可以看出,本文改进的算法具有明显的优势,并且时间开销也比较稳定.

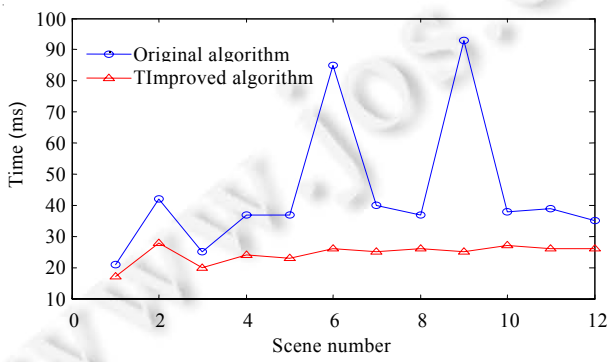


Fig.5 Execution time of PROSAC algorithm

图 5 PROSAC 算法的执行时间

1.1.2 建立 2d-3d 映射表

为了在特征点跟踪过程中更好地通过图像帧中的 2D 特征查找对应场景的 3D 点,需要对重建好的场景 3D 点和对应关键帧的 2D 特征建立映射表.由于每个重建好的 3D 点可能在多个关键帧上都有对应的 2D 特征,则映射表中只存储关键帧 ID、特征点序号和对应三维点序号,以减少实际内存空间的占用,如图 6 所示.

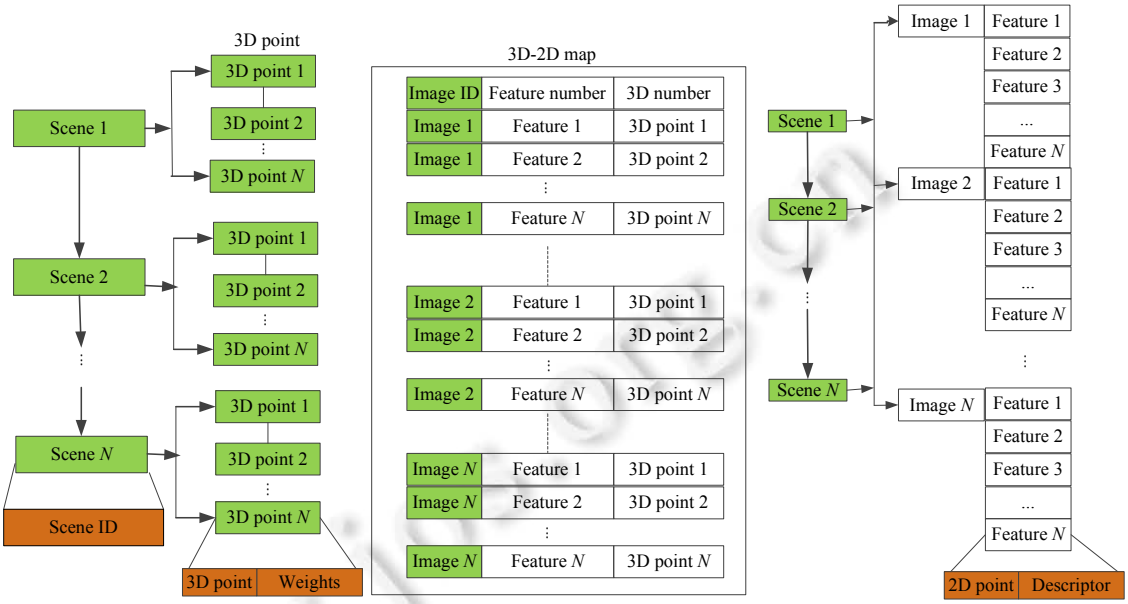


Fig.6 2D-3D map table

图 6 2D-3D 映射表

图 6 中,已知图像的 ID 号和特征点在图像中的存储序号,就可以得到三维点的存储位置,通过存储位置就可以访问场景三维点信息,包括 3D 点(3D point)空间坐标和 3D 点权值(weight),3D 点权值在下一节中通过 SVM 训练得到.2D 特征(feature)包括了二维点在图像中的坐标(2D point)和 64 维描述符(descriptor).在映射表建立完毕后,需要对场景结构行存储,即,将所有重建好的三维点及其对应的特征点描述符加上与跟踪相关映射表都被存至文件中,以供在线注册阶段使用.离线场景重建阶段可以重建多个场景,在线阶段只要加载对应场景结构,即可进行摄像机位姿初始化及摄像机追踪.

1.1.3 训练场景模型 3D 点的权值

为了计算正确的场景模型到跟踪图像的变换关系,需要确保模型的 3D 点与跟踪图像的 2D 点准确地配对,正确配对的程度越高,场景模型到图像的变换关系就越准确.常用的方法如公式(7)所示,通过不断循环,选择得分最高或者匹配点数最多的变换矩阵,作为正确的变换矩阵 P:

$$P=\arg\max_{P'\in T} S(M,I,P') \tag{7}$$

其中,M 代表场景模型,I 代表图像,P' 代表场景模型到图像的变换矩阵,T 代表所有变换矩阵的集合,S 代表得分函数,P 代表分值最大的变换矩阵.

该方法要计算所有的变换矩阵,时间代价非常大,几乎不可行.本文通过对模型的每个 3D 点都设置一个初始的权值来计算分值最高的变化矩阵,采用类似于 PROSAC 的方法来设置循环次数,并以得分值不再增加作为终止条件,避免计算所有的变换矩阵.具体通过公式(8)~公式(11)来实现:

$$S(C,P)=\sum_{(X_j,x_k)\in C} E(\| x_k-P(X_j)\|_2<\tau) \tag{8}$$

$$S_w(C,P)=\sum_{(X_j,x_k)\in C}E(\|x_k-P(X_j)\|_2<\tau)=\langle w,L(C,P)\rangle \tag{9}$$

$$L_j(C,P)=\begin{cases}d_k, \exists(X_j,x_k)\in C:\|x_k-P(X_j)\|_2<\tau\\ 0, \text{ otherwise}\end{cases} \tag{10}$$

$$L(C,P)=[L_1(C,P),L_2(C,P),\ldots,L_J(C,P)]^T L_j,\ 1\leqslant j\leqslant J \tag{11}$$

设图像的 2D 坐标点为 $I=\{x_1,x_2,\ldots,x_k\}$,相应的描述符为 $D=\{d_1,d_2,\ldots,d_k\}$,对应的场景特征点的三维点为 $M=\{X_1,X_2,\ldots,X_k\}$,C 为匹配集合,$C=\{(X_j,x_k,s_{jk})|X_j\in M,x_k\in I,s_{jk}\in R\}$,$s_{jk}$ 为它们的匹配分数,R 是分值集合.

场景模型 3D 点对应的权值为 $w=[w_1,w_2,\ldots,w_J]^T$,得分最大的 P 投影矩阵作为最好的当前图像相对空间特征点投影矩阵,对采集的 10 幅图像都进行学习,对表现比较突出的设置较高的权值.$L(C,P)$为场景模型 3D 点匹配的图像 2D 点对应的描述符.

场景模型 3D 点的权值由样本图像训练得到,使用文献[23]中的方法对给定的训练样本 $\{(I_i,P_i)\}_{i=1}^N$ 进行学习生成权值 W.并根据 W 权值对每个样本图像 I_i,尝试最大程度让真实的 P_i 与其他可供选择的变换矩阵的得分区分开来,如公式(12)、公式(13):

$$\min_{w,\xi}\frac{\lambda}{2}\|w\|^2+\sum_{i=1}^N\xi_i \tag{12}$$

$$\begin{cases}\text{s.t. } \forall i:\ \varepsilon_i\geqslant 0\\ \forall i,\ \forall P\neq P_i:\ \delta S_w^i(P)\geqslant \Delta(P_i,P)-\xi_i\\ \delta S_w^i(P)=S_w(C_i,P)-S_w(C_i,P_i)\end{cases} \tag{13}$$

λ是平衡因子,对训练集精度和权值向量正则化进行权衡.$\Delta(P_i,P)$是损失函数,用来惩罚选择错误的 P 代替正确的 P_i,需要接受的惩罚如公式(14)所示,使用相差的内点数为付出的惩罚代价:

$$\Delta_I(P,P')=|S(C,T)-S(C,T')| \tag{14}$$

1.1.4 训练 Flann 决策树

在线注册时,首先就是要进行场景识别,确定用户当前所在场景,再对场景进行在线学习和跟踪.场景识别就是用户采集当前图像与样本库中的场景图像进行比较,将最相似的样本图像对应的场景确定为当前用户所处场景.最简单的图像识别方法是穷举法,将当前图像与样本图像逐一地进行比较,根据匹配的点数确定最相似的图像.这种方法在样本图像数目较少时可以采用,在样本数量较大时需要耗费大量的时间.本文样本数量已经超过 100 张,显然不能用这种方法.文献[24]对一些相关的算法进行了比较,发现对于高维空间中的最近邻搜索问题,采用分层 K 均值树(hierarchical K-means-tree)和多重随机 KD 树(KD-tree)具有较好的性能,并且实现了根据用户输入的准确度和高维数据自动化选择的近似快速最近邻搜索算法 FLANN,搜索速率得到了显著提高.因此,为了更快速地进行特征匹配,本文选用了 FLANN,它提供了一个快速查找最近似、最近邻的开放源代码库,包括了随机 KD-Tree 和分层 KMeans-Tree 算法,非常适合于在大数据量高维特征中查找最近邻.它会根据样本训练描述符本身的特性,自动选择随机 KD-Tree 或分层 KMeans-Tree 算法进行聚类,提高最近邻匹配的性能.本文将 10 个场景的 120 幅图像提取 SURF 描述符,用 FLANN 算法进行训练,生成 FLANN 决策树,并设置 4 棵 KD-Tree 树进行并行查找,以加快特征匹配速度.再利用 FLANN 进行近似 k 近邻查找,选择 k=2,也就是查找最近邻和次近邻,再以两者的比值来判断是否为匹配点,阈值选择为 0.8.后续通过实验验证 FLANN 算法的匹配性能.

1.2 在线学习与跟踪注册

在线阶段的最终目的是进行场景增强.该阶段需要载入离线阶段提供的场景结构、映射表及关键帧描述符等信息到内存,如果场景信息已经在内存中,就不要再载入了.主要包括以下步骤:

(1) 获取当前图像,提取 SURF 特征描述符.

(2) 用 Flann 决策树进行场景识别.

(3) 对识别成功的场景图像,进行在线学习,建立图像特征点 2D 坐标点到场景三维空间坐标点的对应关系,根据三维矩阵计算得出摄像机的初始位置的姿态矩阵 T_{ck}.

(4) 再用光流对后续帧的进行跟踪,对跟踪到的场景图像进行在线学习,建立后续帧特征点 2D 坐标点到场景三维空间坐标点的对应关系,根据三维矩阵计算得出当前摄像机相对于初始位值的姿态矩阵 T_{kp}.

(5) 用两个姿态矩阵合成注册矩阵 T_{cp}.

(6) 当光流跟踪到的特征点少于设定阈值时,需要进行特征点恢复或者重新进行场景识别.

1.2.1 当前图像帧获取

通过外置的摄像头获取 320×240 像素的 YUV420 连续帧,这些帧被转化为 8 比特每像素的灰度图像与 RGB 图像.黑白图像用来进行场景识别及虚实注册,RGB 图像用来进行场景增强后的结果显示.用 OpenCV2.4.7 库中的 SURF 算法提取当前图像的特征.

1.2.2 场景识别

场景识别就是在样本库中查找当前图像最相似的样本图像,再根据当前图像与样本图像的匹配点数决定是否识别成功.本文使用 Flann 方法来进行特征点匹配,根据匹配点数目确定相似的样本图像,再通过设定最少匹配点阈值和 PROSAC 几何一致性点阈值来判断当前图像与样本图像是否真的匹配.具体实现如下:将当前图像的所有特征点用 Flann 方法查找匹配特征点,Flann 决策树已经在离线阶段通过第 1.1.4 节中的方法生成;再统计所有关键帧图像的特征点匹配数目,选择匹配点数最多的关键帧图像作为相似图像;最后,通过判断匹配点数目是否大于最少匹配点阈值(通常设置为 60~80,根据纹理丰富程度)和用 PROSAC 对匹配点进行几何一致性校验后剩下的匹配点数是否大于设定几何一致性点的阈值(通常设置为 20~30,根据具体图像匹配点数目)来确定当前图像与关键帧图像是否匹配.如果当前图像与关键帧图像匹配点数目满足设定的两个阈值,则当前图像与关键帧图像匹配成功.关键帧图像对应的场景,就是当前拍摄图像所要识别的场景.根据场景 ID,查找场景的 3D 信息,进行后续的在线学习.

本文在 UKBENCH 数据集上测试了 Flann 方法的识别性能.分别选取 50,100,150,200 个场景,每个场景由 4 张图像作为样本训练图像,部分样本图像如图 7(a)所示.模拟实际的场景识别环境,查询图像通过用摄像头拍摄样本图像生成,部分查询图像如图 7(b)所示.图像分辨率统一设置为 320×240,测试的硬件环境与后文第 2 节一样,测试不同场景数上的识别时间和识别精度.识别时间从描述符匹配到查找出相似图像的时间.识别精度为正确识别的场景数与查询图像总数目的比值.若查询图像与相似图像对应同一场景,则是正确识别;否则是错误识别.实验测取 5 次测试结果的平均值.

(a) 训练样本图像集 (b) 查询图像集

Fig.7 Training image set and query image set

图 7 训练图像集与查询图像集

表 1 是随机抽取 4 张样本图像提取特征,测试特征点提取数目、特征点检测时间和描述符计算时间.从表中可以得出:对于纹理较深的图像,提取的特征点数目会超过 400 个;纹理较浅的图像会提取 200 个特征点.同时,特征点的检测和描述符计算时间都较短,都少于 40ms,所以比较适合实时的增强现实应用.

Table 1 Experimental result in detecting features

表 1 特征点检测结果

Image number	Total keypoints	Detect keypoints time (ms)	Calculate descriptors time (ms)
Image 1	507	34.62	30.24
Image 2	260	28.87	18.07
Image 3	471	30.28	28.93
Image 4	163	25.12	15.55

表 2 是对样本图像的提点总数、提点时间和计算描述时间的统计.从表 2 可以看出,在最大样本数为 800 张、提点数目为 307 973 时,特征点检测和计算描述符的总时间为 44 862.8ms,少于 1 分钟,这相对于离线环境来说可以忽略不计.

Table 2 Experimental results in detecting features of all sample images

表 2 所有样本图像特征检测结果

Total images	All keypoints	Detect keypoints time (ms)	Calculate descriptors time (ms)
200	90 883	5 941.29	5 383.92
400	188 210	13 136.1	10 950
600	237 700	19 814.5	14 167.5
800	307 973	26 434	18 428.8

图 8 给出了不同场景数上 Flann 的训练实时间和匹配时间.从图 8(a)可以看出,随着样本数量的增加,训练时间也在增长;但是匹配时间却没有明显的增加,从识别样本数 200 张~800 张仅增加 3.8ms.从图 8(a)可以看出:虽然训练时间随着样本数量增加也在逐渐增长,但是在最大样本数为 800 时,训练时间为 4 339.04ms,不到 1 分钟,这对于离线训练的环境来说还是非常快的.图 8(b)显示了 Flann 的匹配算法是非常实时的,即使在 800 张样本图像,匹配时间也为 13.047ms.图 9 是 Flann 方法的识别精度.从图 9 可以看出,200 张样本图像时识别率达到了 93%,虽然随着样本数量的增长,识别精度有所下降,但是在最大样本数量 800 张时,识别精度仍然保持在 74%.这对于非原始图像的识别来说,已经能够满足常见的增强现实的应用.

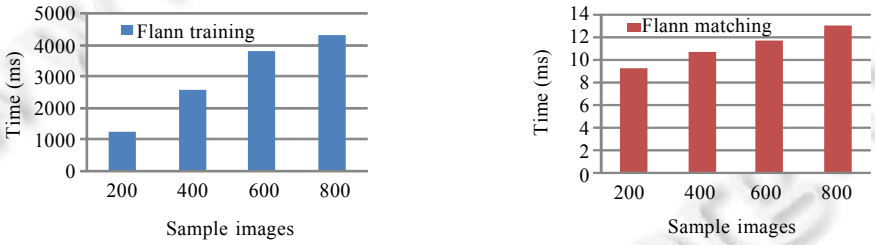


Fig.8 Training time and match time of the Flann method

图 8 Flann 方法的训练时间与匹配时间

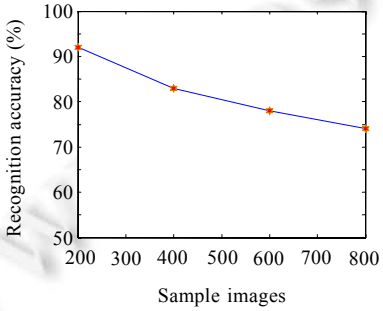


Fig.9 Average recognition accuracy on searching a image

图 9 查询图像平均识别精度

1.2.3 场景在线学习

在跟踪过程中,需要对场景进行在线实时的学习来更新场景 3D 点的 w 权值,以适应不断变换的场景环境.场景 3D 点的 w 初始值只能用于静态的训练图像计算的分值最高的变换矩阵 P,在线环境下需要不断地对场景进行学习来更新 w 值,以适应跟踪过程中不断变换的外部场景.场景学习方法包括:首先,使用第 1.1.1 节的图像匹配方法对当前图像与已识别出场景的关键帧图像进行匹配,确定匹配点对;其次,根据场景和关键帧图像特征点信息查找 2D-3D 映射表,确定场景的 3D 点与当前图像的 2D 特征点的对应关系;然后,对关键帧图像与当前图像的匹配点对执行 RANSAC 操作去除误匹配点,对 RANSAC 的中间结果,使用第 1.1.3 节中的方法计算得分,选择得分值最大的匹配点对,形成匹配点对〈场景 3D 点,当前图像 2D 点〉,传递给后文第 1.2.4 节,用来计算三维注册矩阵;最后,更新 w 值,使其适应后续的变化环境.

w 值的更新使用文献[25]中的方法,对场景的每个 3D 点,计算当前图像中与场景 3D 点的变换矩阵,选择得分最高和次高的变换矩阵,并用这两个分值去更新权值 w,如公式(15)~公式(19):

$$w_j^{t+1} \leftarrow (1-\eta_t\lambda)w_j^t + E(\max_{P\neq P_t}\{\Delta(P_t,P)-\delta S_w^i(P)\}>0)\eta_t\alpha_j^t + E(u_j\in C_t^*)E(\max_{k'\neq k}\{1-\langle w_j, d_k-d_{k'}\rangle\}>0)\eta_t\nu\beta_j^t \tag{15}$$

$$\alpha_j^t = L_j(C_t,P_t) - L_j(C_t,\hat{P}) \tag{16}$$

$$\beta_j^t = d_k - d_{\hat{k}} \tag{17}$$

$$\hat{P} = \arg\max_{P\neq P_t}\{\Delta(P_t,P)-\delta S_w^i(P)\} \tag{18}$$

$$\hat{k} = \arg\max_{k'\neq k}\{1-\langle w_j, d_k-d_{k'}\rangle\} \tag{19}$$

j 代表场景的 3D 点序号;t 代表当前帧帧号;w_j^{t+1} 为下一帧的的权值;w_j^t 为当前帧二维特征点对应场景第 j 个 3D 点的权值;P_t 代表当前帧得分最高的变换矩阵;$\eta_t=1/\lambda t$ 代表步长,λ是平衡因子,权衡训练集精度和权值向量正则化.

直接使用文献[25]中的 w 值更新方法,需要计算当前图像存在的所有变换矩阵的得分,并选取得分最高与次高的分值,运用公式(15)进行更新,计算量较大.本文对该方法进行改进,降低单独计算分值的开销,直接利用 RANSAC 算法的中间结果计算最大值和次大值,进行 w 值更新.最大值由 RANSAC 算法结束时剩下的内点和对应的 3D 点根据公式(9)得出近似最大值代替,次大值由 RANSAC 的中间得分仅小于近似最大值的分值代替.同时,为了减少 RANSAC 算法的时间开销和提高去除误匹配点的性能,文中对 RANSAC 算法进行如下 3 条规定:(1) 对当前图像特征点按分值进行降序排列(分值为特征点描述符乘以对应场景 3D 点的权值),RANSAC 算法按分值从高到低选择图像特征点;(2) 选取的图像特征点不能在一条线上;(3) 选取的图像特征点距离不能太近,否则 RANSAC 算法效果不好,尽量使选择的特征点分布比较均匀.

1.2.4 三维注册矩阵计算

在得到当前图像特征点 m_t 与其三维空间坐标点 M_t 的对应关系后,在摄像机内部参数 K 已知的情况下,即可采用 P-N-P[26]算法求出当前摄像机的位姿矩阵 $T=[R|t]$,如下式:

$$\tilde{m} = \lambda K[R\,|\,t]\tilde{M} \tag{20}$$

R 代表旋转矩阵,t 代表平移矩阵.为了提高参数估计的精度,强化算法对错误数据匹配的容忍度,我们采用 Tukey M-估计算法计算摄像机的位姿.M-估计[27]是利用最小化误差残 $\min_T\sum_{i=1}^{n}\rho(r_i)$ 获取参数的最优估计.此处,σ为连续对称函数,$r_i=\|m_i-\lambda KTM_i\|$为图像的反投影误差.Tukey M-估计算法中的$\sigma$用以下函数表示:

$$\begin{cases}\sigma(x)=\dfrac{c^2}{6}\left[1-\left(1-\left(\dfrac{x}{c}\right)^2\right)^3\right], & \text{if } |x|\leqslant c\\ \sigma(x)=\dfrac{c^2}{6}, & \text{if } |x|>c\end{cases} \tag{21}$$

通过迭代优化,即可求解出摄像机的旋转和平移矩阵.此处,M-估计的初始参数值 c 为可以调整的常量,用于

调整算法的敏感度,本文选择所有残差的均方差作为其初始值.

1.2.5 虚实融合显示

三维注册的最终目的就是将虚拟物体准确地叠加在真实场景中,实现无缝的虚实融合显示.实现虚拟物体的正确叠加,就需要获取当前摄像机坐标系与世界坐标系之间的坐标变换矩阵 T_{cp}.T_{cp} 通过已知的摄像机平面坐标系与世界坐标系之间的初始姿态矩阵 T_{cp} 和当前摄像机相对初始位置的注册矩阵 T_{ck} 合成得到,如下式:

$$T_{cp}=T_{ck}\cdot T_{kp} \tag{22}$$

有了当前摄像机的注册矩阵,就可以利用 OpenGL(open graphics library)实现对虚拟物体进行映射变换,叠加到场景正确的位置.若需要叠加虚拟物体是二维图像,则直接以贴纹理的方式实现;若是 3D 模型,则载入 OBJ 模型文件,以模型中的组(group)为单位,每个组都有自己对应的材质,分别在内存中存储其点、线、面、材质和纹理信息,并分组绘制和贴纹理.

1.2.6 光流跟踪

在增强现实的三维注册过程中,如果每次都对整幅图像进行特征点提取和匹配来计算机摄像机的姿态,会耗费大量的计算时间,使算法的实时性变差.在实际应用中,用户在一个场景的运动都是比较平缓的,不会剧烈地晃动摄像机或者突然大幅度移动摄像机,摄像机的姿态在相邻帧之间变化不会太大;通过相邻帧之间的连续性,可以有效地估计特征点在下一帧中出现的位置,从而快速计算摄像机的姿态.本系统采用光流预测的方法预测下一帧的图像特征点的坐标.以减少反复地对整幅进行特征提取和识别的时间,加快特征点跟踪速度,提高算法的实时性.

当计算出当前图像帧的摄像机姿态以后,后续帧通过光流来计算特征点的坐标,进而建立起后续帧特征点 x_{t+1} 与其三维空间坐标点 x_{t+1} 的对应关系,通过 P-N-P 算法计算摄像机的姿态.光流跟踪利用图像序列中的像素强度的时域变化和相关性来确定像素点的“运动”.光流算法[28]基于如下假设:相邻帧之间亮度恒定,相邻帧之间目标运动比较微小.用 $I(x,y,t)$表示 t 时刻像素点(x,y)的灰度值,如公式(23):

$$I(x,y,t)=I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t) \tag{23}$$

使用泰勒级数展开:

$$I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t)=I(x,y,t)+\frac{\partial I}{\partial x}\mathrm{d}x+\frac{\partial I}{\partial y}\mathrm{d}y+\frac{\partial I}{\partial t}\mathrm{d}t \tag{24}$$

即,$I_x\mathrm{d}x+I_y\mathrm{d}y+I_t\mathrm{d}t=0$,令 $u=\frac{\mathrm{d}x}{\mathrm{d}t}$,$v=\frac{\mathrm{d}y}{\mathrm{d}t}$,可得:

$$I_xu+I_yv=-I_t \tag{25}$$

光流算法基于上述公式计算特征点像素的“运动”.实验结果表明,使用光流算法计算相邻帧之间特征点的坐标变化用时只需几十毫秒.随着时间的推移,光流跟踪上的点数将越来越少,进而影响摄像机姿态的计算精度.本文在光流点数小于设定的阈值 T_1(本文为保证注册矩阵准确性,T_1 设为 30)的情况下,对当前帧提取特征点进行匹配,同时,如果连续三帧都匹配失败,则说明用户已经离开了当前场景,进入到新的场景,这样,需要重新拍摄关键帧进行新的场景重建.如果光流跟踪的点数少于设定阈值 T_2(本文为保证注册矩阵的精度,T_2 设为 40),则需要进行丢失的特征点恢复.

1.2.7 特征恢复

光流跟踪本身容易产生漂移的问题,随着摄像机长时间的移动,光流跟踪到的点数也越来越少.为了保证计算摄像机注册矩阵的精度,在跟踪到点的少于设定阈值 T_2=40 时,进行丢失的特征恢复.文献[3]提出了通过关键帧与当帧的单应关系来恢复丢失特征的方法.该方法首先进行图像识别,选择属于同一场景匹配分值最高的 3 个关键帧,再计算 3 个关键帧与当前图像的单应矩阵;然后,通过单应矩阵预测 3 个关键帧丢失的特征在当前图像的位置;最后,在当前图像预测位置周围±20 像素区域里执行 SSD(sum of squared difference)方法查找与关键帧匹配的特征,对其进行恢复.文献[3]中的实验结果表明:使用该方法恢复的特征点比较准确,但是需要重新进行图像识别,增加了计算量,影响了跟踪注册的实时性.文献[9]提出了通过单个关键帧和投影矩阵来恢复丢失的

特征的方法.该方法首先利用投影矩阵预测丢失特征点在当前图像中的位置;再通过关键帧与当前图像的单应关系,将丢失的特征在关键帧上像素块变换到当前图像;然后,在当前图像的预测位置周围±10 像素区域里检测特征;最后,对被检测到的特征与变换到当前图像的特征块执行 NCC(normalized cross-correlation)操作,选择 NCC 分值最高的特征进行恢复.文献[9]中实验结果表明,该方法恢复特征速度较快,但是 NCC 方法本身存在对场景光照变化、部分遮挡或变形较敏感,鲁棒性较差.

所以,本文为了能够快速恢复当前图像的特征和提高恢复的特征对环境具有的鲁棒性,不再对跟踪图像进行识别,只有在跟踪失败时才执行识别算法,重新初始化特征点;使用对光照和旋转具有鲁棒性的 SURF 特征来恢复丢失的相似特征.

- 首先,利用当前图像上已跟踪到关键帧的特征,得出当前图像相对于关键帧的单应关系 H_k^c,如下式:

$$\begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} = H_k^c \begin{bmatrix} x_k \\ y_k \\ 1 \end{bmatrix} \tag{26}$$

  $(x_k,y_k,1)$为关键帧特征点坐标,$(x_c,y_c,1)$为对应当前图像跟踪到的特征点坐标.
- 其次,通过单应关系预测当前图像丢失特征的位置,设 f 关键帧上的特征点,坐标为$(f_x,f_y,1)$,在当前图像上已经丢失,通过 H_k^c 来预测其在当前图像的特征点位置为 $(f_x',f_y',1)$.
- 然后,在当前图像预测位置的周围±10 像素区域检测 SURF 特征点.
- 最后,将当前图像检测到的特征点与特征 f 进行比较,选择与 f 距离最近的特征点 $f_{\min}$,并判断在特征 f 周围±10 像素区域 f 与 $f_{\min}$ 距离是否仍旧是最短的:如果是,则恢复该特征点;否则丢掉.

当需要恢复的特征点数目较多时,恢复时间也较长.为保证跟踪算法的实时性,设置特征点恢复数目,当恢复数目到达设置的值(本文设置为 80)时不再进行恢复,继续后续帧的光流跟踪步骤.

2 实验与分析

本文的实验设备包括一台个人普通计算机(personal computer)和一个普通摄像头(camera),其参数如下:PC 参数为处理器 Intel® Core(TM) i5,CPU 频率为 2.67GHZ,操作系统 WIN7,内存 8G;Camera 为手持摄像头,型号为天敏(D805HD),CMOS 类型,最高分辨率 4992×3328,最大帧数 60 帧/秒.摄像头的内参通过文献[15]中的方法进行标定.实验数据使用 UKBENCH 数据集 5 个场景和校园里面室外拍的 8 个场景,共 13 个场景作为样本场景,摄像机拍摄的视频帧分辨率统一设置为 320×240.对纹理丰富的图像,提取特征点数目超过 400 个,则根据特征点的响应强度,只选取响应强度最高的前 400 个特征点.因为特征点数目过多,给后续的匹配和姿态计算增加了许多工作量,同时,跟踪精度并不能得到显著提高.实验中,渲染的虚拟模型采用 3DS MAX 进行建模设计,统一导出为 OBJ 格式的 3 维模型文件.实验目的是,测试本文算法的跟踪效果以及跟踪的精度和速度.跟踪精度使用 RMS(root mean square) errors(实际像素点坐标与预测的点坐标的平方根差)[29]来测试;跟踪速度用帧率(每秒能够处理的帧数)来衡量.

2.1 算法的跟踪效果

本文在不同的视角、不同的距离和不同光照条件下测试了算法的鲁棒性.在图 10(a)~图 10(g)中,系统在不同自然环境下,即使在部分场景光照较暗或者摄像头沿不同坐标轴旋转的情况下,也可以准确、实时地完成注册.图 10 的左半部分是对校园拍摄场景的注册,右半部分是对 UKBENCH 库中场景的注册.图 10(a)~图 10(g)是在光线比较亮的环境下,分别在不同的视角、不同距离移动和缩放的实验效果图,图 10(h)~图 10(m)是光线较明亮场景的实验效果图,图 10(n)是在场景部分遮挡的情况下进行注册的效果图.可以看出,本文的算法能够准确地、较为鲁棒地完成注册.

(a) 摄像机沿*Z*轴旋转 (b) 摄像机沿*Z*轴旋转

(c) 摄像机沿*Y*轴旋转 (d) 摄像机沿*Y*轴旋转

(e) 摄像机沿*X*轴旋转 (f) 摄像机沿*X*轴旋转

(g) 摄像机沿*X*轴平移 (h) 摄像机沿*X*轴平移

(i) 摄像机沿*Z*轴缩放 (j) 摄像机沿*Z*轴旋转

(k) 摄像机沿*X*轴旋转 (l) 摄像机沿*Y*轴旋转

(m) 摄像机沿*Z*轴旋转与缩放 (n) 摄像机部分遮挡

Fig.10 Tracking and register result

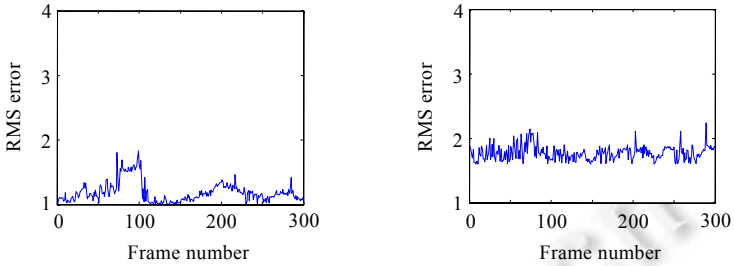
图 10 跟踪注册结果

2.2 算法的跟踪精度

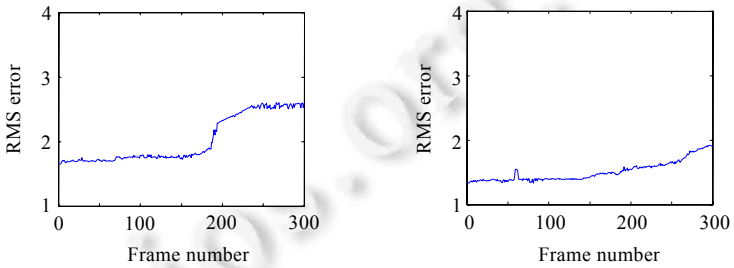
本文使用提出的算法对连续的 300 帧图像进行跟踪,测试图像特征点的真实坐标与用注册矩阵重投影的坐标的 RMS 误差.模拟用户手持摄像头的运动方式,摄像头沿着 *X*,*Y*,*Z* 轴平缓地旋转,模拟用户不同的视角或者手拿摄像机是否倾斜:沿着 *Z* 轴移动模拟人由远到近放大场景,分别测试 4 种方式下图像特征的 RMS 误差.选取 100 个特征点计算在图像中的真实坐标与重投影坐标误差的平均值,实验结果如图 11 所示.图 11(a)表明,当摄像头沿着 *Z* 轴旋转 0°~60°时,RMS 误差小于一个像素.图 11(b)表明,当摄像头沿着 *Y* 轴旋转 0°~40°时,重投影误差接近 2.5 个像素.图 11(c)表明,当摄像头沿着 *X* 轴旋转 0°~40°时,重投影误差接近 2.8 个像素.图 11(d)表明,当摄像头沿着 *Z* 轴移动由远到近模拟对场景的缩放,近似放大 1 倍时,重投影误差接近 2 个像素.

本文对提出的跟踪注册算法与 Yuan[29],Tao[8]提出的跟踪注册算法在同一场景连续的 150 帧图像进行注册精度测试.实验中不断改变摄像头的视角,0°~40°,选取 100 个特征点计算在图像中的真实坐标与重投影坐标误差的平均值,实验结果如图 12 所示.图 12(a)表明,Yuan[29]的算法随着时间的推移注册误差逐渐增大;而本文的算

法跟踪相对比较稳定,同时注册误差也较低.图 12(b)表明,Tao[8]提出的算法跟踪注册效果比较稳定,注册误差也较低少于 3 个像素;但是本文算法的注册误差要优于 Tao 的算法,平均重投影误差少于 2 个像素.所以,实验结果表明,本文提出的跟踪注册算法具有较高的跟踪注册精度.



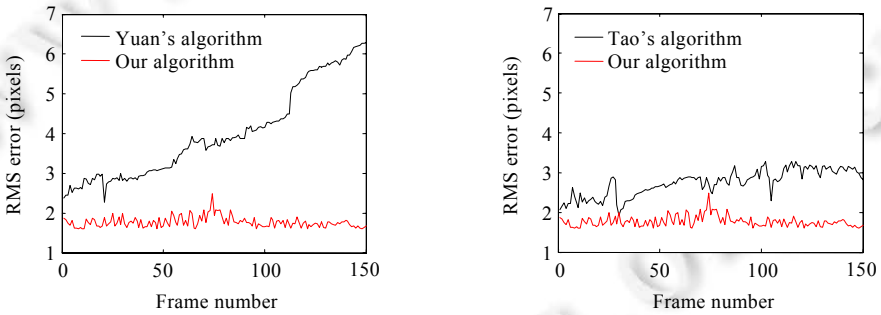
(a) 摄像头绕 Z 轴旋转 0°~60° (b) 摄像头绕 Y 轴旋转 0°~40°



(c) 摄像头绕 X 轴旋转 0°~40° (d) 摄像头沿 Z 轴移动放大场景

Fig.11 RMS error for the proposed trackingalgorithm

图 11 本文跟踪算法的 RMS 误差



(a) 本文算法与 Yuan[26]提出的方法对比

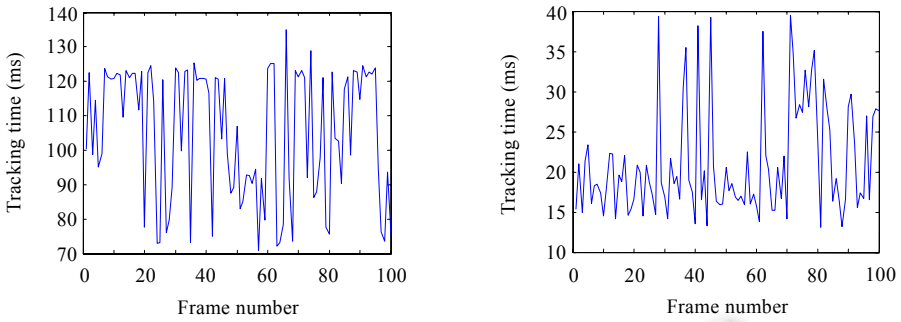
(b) 本文算法与 Tao[8]提出的方法对比

Fig.12 RMS error for the proposed algorithm vs. Yuan's algorithm[26] and Tao's algorithm[8]

图 12 本文算法与 Yuan 的算法[26]和 Tao 的算法[8]的跟踪误差对比

2.3 算法的跟踪时间

在本文描述的跟踪算法中,使用光流算法跟踪到的特征点计算摄像头的姿态.当目标丢失时,通过 SURF 提点算法计算摄像头的初始化姿态.SURF 提点算法与光流算法的时间开销,影响整个跟踪系统的实时性能.对连续的多帧图像,分别使用提点匹配的跟踪方法和光流跟踪方法,统计两种方法每一帧图像的处理时间.实验结果如图 13 所示,提点跟踪的算法时间开销在 70ms~140ms 内波动,光流跟踪算法时间开销在 40ms 左右.因此,从图 13 可知,提点匹配方法的跟踪速度最坏情况不到 8f/s,实时性较差;而光流的跟踪速度能够达到 25 帧/秒,实时性较好,跟踪过程也较为流畅.



(a) 提点匹配方法的跟踪时间 (b) 光流方法的跟踪时间

Fig.13 Time comparison of optical flow tracking approach and detection matching approach

图 13 提点匹配跟踪方法与光流跟踪方法的时间对比

表 3 是对在线跟踪注册过程的各步骤消耗时间的统计.从表 3 可以得出,摄像机初始化时,场景的平均识别时间不到 70ms,摄像机在线跟踪注册总耗时不到 40ms,帧率可以达到 25f/s,可以满足实时的增强现实应用.

Table 3 Computation time details for whole registration process

表 3 整个注册过程时间开销明细

	Step	Time (ms)
Initialize camera pose	SURF detecting	51
	Scene recognition	15
	Scene learning	12
Online tracking and register	Feature tracking	2.4
	Pose calculating	10.3
	Lost feature recovering	25

3 结 论

本文提出了一种基于自然场景在线学习的增强现实跟踪注册算法,算法依据在线场景的视觉信息,获得摄像机的 6 自由度注册矩阵.算法在已知部分三维场景结构以及少量标定关键帧的基础上,实现三维场景的实时跟踪注册.利用机器学习的方法解决 RANSAC 误匹配问题,并对场景的自然特征点进行光流预测,以消除系统注册中的抖动问题.实验结果表明,该算法鲁棒性强,注册精度高,实时性好.

但是,目前本文提出的算法还存在以下问题:

(1) 用户的视角区域有限.当用户处于与被拍摄的参考帧图像相同的位置时效果较好,当用户在其他视角拍摄的图像与参考图像相差较大时,跟踪效果较差.

(2) 实时跟踪的场景数目较少.因为图像描述符的提取和匹配消耗了大量的时间,不能对大规模的场景进行识别和跟踪注册.

将来的工作将利用引入更快速的二进制描述符提取算法和并行处理,以加速二进制描述符汉明距离比较.

致谢 在此,向对本文的工作给予支持和建议的同行,尤其是北京市混合现实与新型显示工程技术研究中心、北京理工大学颜色与信息系统实验室的老师和同学表示感谢.

References:

[1] Liestol G, Morrison A. Views, alignment and incongruity in indirect augmented reality. In: Proc. of the IEEE Int'l Symp. on Mixed and Augmented Reality—Arts, Media and Humanities (ISMAR-AMH). IEEE, 2013. 23−28. [doi: 10.1109/ISMAR-AMH.2013.6671263]

[2] Guan T, Duan LY, Yu JQ, Chen YJ, Chen YJ, Zhang X. Real time camera pose estimation for wide area augmented reality applications. IEEE Computer Graphics and Applications, 2011,32(3):56−68. [doi: 10.1109/MCG.2010.23]

[3] Duan LY, Guan T, Luo YW. Wide area registration on camera phones for mobile augmented reality applications. Sensor Review, 2013,33(3):209−219. [doi: 10.1108/02602281311324663]

[4] Vista IV, Felipe P, Lee DJ, Chong KT. Remote activation and confidence factor setting of ARToolKit with data association for tracking multiple markers. Int'l Journal of Control and Automation, 2013,6(6):243−252. [doi: 10.14257/ijca.2013.6.6.23]

[5] Fiala M. ARTag: An improved marker system based on AR toolkit. National Research Council Publication, 2004,4(7):166−174. [doi: 10.4224/5763247]

[6] Hong ZB, Mei X, Prokhorov D, Tao DC. Tracking via robust multi-task multi-view joint sparse representation. In: Proc. of the IEEE Int'l Conf. on Computer Vision (ICCV). IEEE, 2013. 649−656. [doi: 10.1109/ICCV.2013.86]

[7] Srinath S, Antti O, Christian T. Interactive markerless articulated hand motion tracking using RGB and depth data. In: Proc. of the IEEE Int'l Conf. on Computer Vision. IEEE, 2013. 2456−2463. [doi: 10.1109/ICCV.2013.305]

[8] Guan T, Duan LY, Chen YJ, Yu J. Fast scene recognition and camera relocalisation for wide area augmented reality systems. Sensors, 2010,10(6):6017−6043. [doi: 10.3390/s100606017]

[9] Guan T, Wang C. Registration based on scene recognition and natural features tracking techniques for wide-area augmented reality systems. IEEE Trans. on Multimedia, 2009,11(8):1393−1406. [doi: 10.1109/TMM.2009.2032684]

[10] Basha T, Avidan S, Hornung A, Matusik W. Structure and motion from scene registration. In: Proc. of the 2012 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). IEEE, 2012. 1426−1433. [doi: 10.1109/CVPR.2012.6247830]

[11] Lim H, Sinha SN, Cohen MF. Real-Time image-based 6-DOF localization in large-scale environments. In: Proc. of the 2012 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). IEEE, 2012. 1043−1050. [doi: 10.1109/CVPR.2012.6247782]

[12] Wei BC, Guan T, Duan LY, Yu JQ, Mao T. Wide area localization and tracking on camera phones for mobile augmented reality systems. Multimedia Systems, 2015,21(4):381−399. [doi: 10.1007/s00530-014-0364-2]

[13] Nister D, Naroditsky O, Bergen J. Visual odometry. In: Proc. of the Conf. on Computer Vision and Pattern Recognition. Washington, 2004. 652−659. [doi: 10.1109/CVPR.2004.265]

[14] Bay H, Tuytelaars T, Van Gool L. Speeded up robust features (SURF). Computer Vision and Image Understanding, 2008,110(3):346−389. [doi: 10.1016/j.cviu.2007.09.014]

[15] Zhang ZY. A flexible new technique for camera calibration. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000,22(11):1330−1334. [doi: 10.1109/34.888718]

[16] Delabarre B, Marchand E. Camera localization using mutual information-based multiplane tracking. In: Proc. of the IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2013). IEEE, 2013. 1620−1625. [doi: 10.1109/IROS.2013.6696566]

[17] Lowe DG. Distinctive image features from scale-invariant keypoints. Int'l Journal of Computer Vision, 2004,60(2):91−110. [doi: 10.1023/B:VISI.0000029664.99615.94]

[18] Rosten E, Drummond T. Machine learning for high speed corner detection. In: Proc. of the European Conf. on Computer Vision, Vol.1. Berlin, Heidelberg: Springer-Verlag, 2006. [doi: 10.1007/11744023_34]

[19] Ethan R, Vincent R, Kurt K, Bradski G. ORB: An efficient alternative to SIFT or SURF. In: Proc. of the Int'l Conf. on Computer Vision. IEEE, 2011. [doi: 10.1109/ICCV.2011.6126544]

[20] Calonder M, Lepetit V, Strecha C, Fua P. Brief: Binaryrobust independent elementary features. In: Proc. of the European Conf. on Computer Vision. Berlin, Heidelberg: Springer-Verlag, 2010. 778−792. [doi: 10.1007/978-3-642-15561-1_56]

[21] Nister D, Stewenius H. Scalable recognition with a vocabulary tree. In: Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR). IEEE, 2006. 2161−2168. [doi: 10.1109/CVPR.2006.264]

[22] Chum O, Matas J. Matching with PROSAC-progressive sample consensus. In: Proc. of the 2005 Computer Society Conf. on Computer Vision and Pattern Recognition. IEEE, 2005. 220−226. [doi: 10.1109/CVPR.2005.221]

[23] Tsochantaridis I, Joachims T, Hofmann T, Altun Y. Large margin methods for structured and interdependent output variables. Journal of Machine Learning Research, 2005,6(2):1453−1484.

[24] Muja M, Lowe DG. Fast approximate nearest neighbors with automatic algorithm configuration. In: Proc. of the VISAPP. INSTICC, 2009. 331−340.

[25] Hare S, Saffari A, Torr PHS. Efficient online structured output learning for keypoint-based object tracking. In: Proc. of the 2012 Computer Society Conf. on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society, 2012. 1894−1901. [doi: 10.1109/CVPR.2012.6247889]

[26] Lu C, Hager G, Mjolsness E. Fast and globally convergent pose estimation from video images. Trans. on Pattern Analysis and Machine Intelligence, 2002,22(6):610−622. [doi: 10.1109/34.862199]

[27] Meer P. Robust Techniques for Computer Vision. New York: Prentice Hall, 2004.

[28] Philippe W, Jerome R, Zaid H, Cordelia S. DeepFlow: Large displacement optical flow with deep matching. In: Proc. of the IEEE Int'l Conf. on Computer Vision. IEEE, 2013. 1385−1392. [doi: 10.1109/ICCV.2013.175]

[29] Yuan ML, Ong SK, Nee AYC. Registration using natural features for augmented reality systems. IEEE Trans. on Visualization and Computer Graphics, 2006,12(4):569−580. [doi: 10.1109/TVCG.2006.79]

桂振文(1983－),男,湖南祁阳人,博士,高级工程师,主要研究领域为计算机视觉,图像处理,移动增强现实.

陈靖(1974－),女,博士,副教授,CCF 专业会员,主要研究领域为模式识别,移动增强现实.

刘越(1968－),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为虚拟现实,增强现实.

王涌天(1957－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为虚拟现实,增强现实.