

中图法分类号: TP391.9 文献标志码: A 文章编号: 1006-8961(2010)12-1842-07

论文索引信息: 杨明浩, 王阳生. 保持视觉稳定性的增强现实注册算法[J]. 中国图象图形学报, 2010, 15(12): 1842-1848

# 保持视觉稳定性的增强现实注册算法

杨明浩, 王阳生

(中国科学院自动化研究所, 北京 100190)

**摘要:** 提出了一种保持视觉稳定性的增强现实注册算法, 特征点在视频中的初匹配基于随机树检索统计得到, 初步匹配的特征点采用半几何限制结合单应矩阵迭代的方法进行错误匹配排除, 随后采用光流法跟踪特征点, 最后根据共面 POSIT 算法计算出物体世界坐标系到摄像机坐标系的变换矩阵。由于采用了特征点跟踪方法, 在一定程度上有效的防止了虚拟物体在视觉上的抖动。

**关键词:** 增强现实; 随机树; 光流法

## Visual robust register algorithm for augmented reality

YANG Minghao, WANG Yangsheng

(Institute of Automation, Chinese Academy of Sciences, Beijing 100190)

**Abstract:** A visual robust augmented reality (AR) register algorithm based on retrieving and tracking pre-defined image feature points is given in this paper. Feature points are retrieved from video by voting from random tree. False matching points are removed by semi local constraint paradigm with iterative homograph constraint. Optical flow is used to track the pre-defined tracking points. Rotation and translation matrix from world coordinate to camera coordinate are obtained with coplanar POSIT algorithm. For tracking method being used in our algorithm, dithering problem is effectively solved in real-time experience.

**Keywords:** augmented reality; random tree; optical flow

### 1 视频增强现实的视觉稳定性

增强现实技术(AR)<sup>[1]</sup>是虚拟现实技术一个重要分支,它利用计算机产生的附加信息对真实场景进行增强,并借助感知和显示设备将虚拟和真实信息进行融合,呈现出比单一虚景,单一实景蕴含更丰富信息的新场景。增强现实技术在医疗、航天、工业制造、教育、娱乐等领域具有广泛的应用前景,受到研究者越来越多的关注。

视频增强现实是指利用摄像头采集到的视频数据作为真实场景数据来源,通过分析视频内容,把计

算机生成的虚拟物体合成到视频流中,这个过程称为 3 维注册。提出的方法主要针对虚拟物体的视觉稳定性进行,视频增强现实视觉稳定性分为 2 个方面,即几何位置的准确性和连贯性。准确性在几何方面指人眼借助显示设备观测虚实结合的场景时,虚拟物体必须准确的放置于真实场景中,连贯性指虚拟物体在上下视频帧间保持不抖动<sup>[2]</sup>。其中位置准确性的关键在于精确的 3 维注册,即需要得到准确的物体坐标系到摄像机坐标系的旋转和平移矩阵(后文将旋转平移矩阵合称为变换矩阵)。虚拟物体在上下视频帧间的不抖动性关键在于要保证视频上下帧间虚拟物体位置的不漂移。

基金项目:北京市海淀园文化创意产业基金项目(2007-CY-03)。

收稿日期:2009-04-20; 改回日期:2009-09-07

第一作者简介:杨明浩(1977—),男。中国科学院自动化研究所博士研究生。研究方向为计算机图形学、计算机视觉、计算机动画技术。E-mail:lingau\_doc@163.com。

ARToolkit<sup>[3]</sup>通过视频图像二值化,对黑色区域进行边缘分割,模板匹配等方法找到黑色标记的 4 个顶点,计算出变换矩阵。Lepetit<sup>[4]</sup>采用随机树方法<sup>[5]</sup>检测特征点,依照 RANSAC<sup>[6]</sup>最大可能排除错误匹配点,然后采用  $n$  点透视的方法计算出变换矩阵。这 2 种方法借助检测实现注册,不需要对标记的位置进行初始化,但由于光照变化和图像噪声的影响,场景静止时,上下帧间虚拟物体在视觉观测上并不稳定。Simon<sup>[7]</sup>采用跟踪平面特征点,在帧间选取最小误差点计算变换矩阵。机器伺服技术<sup>[8-9]</sup>通过控制眼手一体的摄像机,跟踪场景中的物体,计算得到摄像机和物体之间的变换矩阵。后面两种方法因为跟踪方法的引入,能够保证视频中真实场景静止时,虚拟物体的不抖动,但虚拟物体初始(即初始跟踪)位置不能确定。受文献[4,7]的启发,提出了一种保持视觉稳定性的增强现实注册算法,预定义特征点在视频中的初匹配基于随机树投票得到,初步匹配的特征点采用半几何限制结合单应矩阵迭代的方法进行错误匹配排除。随后采用光流法跟踪特征点,最后根据共面 POSIT<sup>[10-11]</sup>算法计算出物体世界坐标系到摄像机坐标系的变换矩阵。实时增强视频序列表明,该方法在一定程度上有效的防止了虚拟物体在视觉上的抖动,特别是视频序列静止时,虚拟物体基本保持稳定。

## 2 特征点选取及检索树的生成

### 2.1 特征点选取

从视频某帧中选取一个矩形范围作为模板图像(图 1(a))。模板中的 Harris 角点<sup>[12]</sup>作为检索特征

点,记为  $T_0$ 。 $T_0$  中具有良好跟踪特性的点<sup>[13]</sup>作为跟踪备选点,记为  $T_1$ ,从  $T_1$  中选取均匀分布的少数点作为跟踪特征点,记为  $T_2$  ( $T_0 \supset T_1 \supset T_2$ )。图 1(b)展示了模板中预定义的特征点,其中小叉表示 Harris 特征点 ( $T_0$ ),空心圆圈表示好的跟踪点 ( $T_1$ ),实心圆点是均匀分布的少量跟踪特征点 ( $T_2$ )。本模板中, $T_0$  和  $T_2$  的个数分别为 179 和 13。

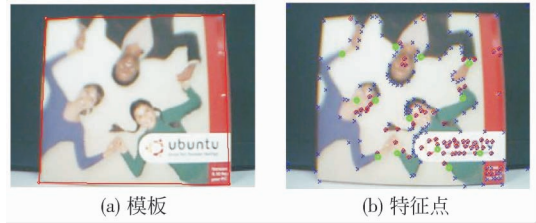


图 1 模板与特征点  
Fig. 1 Template and feature points

### 2.2 检索树的生成

检索树依照文献[4]描述的方法产生:对应于视频中各种角度,不同距离图像中的特征点,首先对模板图像经过仿射变换产生大量图像。设仿射变换为  $A * I$ ,其中  $A = R_\theta R_\phi^{-1} S R_\phi$ ,其中  $R_\theta \in [-\theta, \theta]$ ,  $R_\phi \in [-\Phi, \Phi]$ ,  $S = A(\lambda_1, \lambda_2)$ ,  $A$  是由  $\lambda_1$  和  $\lambda_2$  构成的  $2 \times 2$  对角阵, $\lambda_1, \lambda_2 \in [\Pi_1, \Pi_2]$ ,  $I$  为 2.1 节中选取的模板。产生每个仿射图像时,为了缓解视频图像噪声的影响,加入了高斯白噪声。图 2 显示了几幅模板图像经过仿射变换并添加了白噪声的图像,其中各图  $R_\theta, R_\phi, \lambda_1, \lambda_2$  的取值分别为 (a)  $(-60, 50, 1.0, 1.0)$ , (b)  $(0, 0, 1.8, 0.2)$ , (c)  $(0, 50, 1.0, 0.2)$ , (d)  $(30, -100, 1.4, 1.8)$ , (e)  $(30, 0, 0.2, 1.4)$ , (f)  $(-30, 0, 0.2, 0.2)$ ,非图像部分噪声均值分别为 RGB 通道

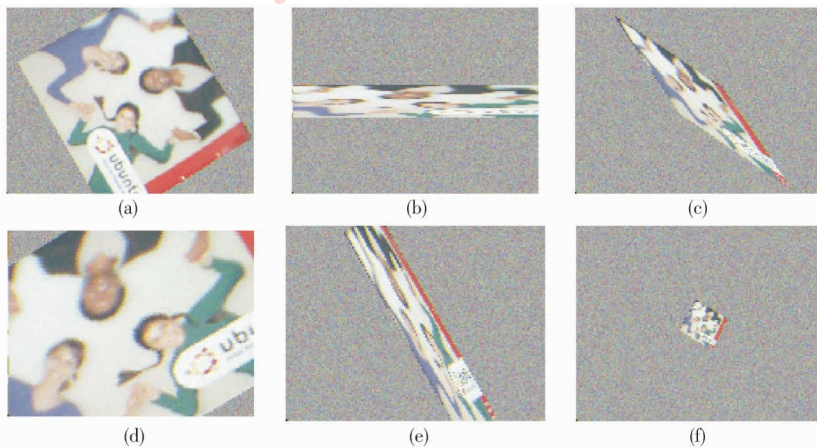


图 2 仿射变化并添加白噪声后的模板图像  
Fig. 2 Affine and adding Gaussian noise templates

的均值,图像部分噪声均值为 0,两部分噪声方差均为 10。

基于上面产生的仿射变换噪声图像,对模板图像中的每个特征点,由  $A[x, y]^T$  得到该点在某仿射图像中的坐标。根据 SIFT<sup>[14]</sup> 特征向量生成办法,把点邻域像素基于主梯度方向旋转并对齐到  $x$  轴,同时采样对齐后像素点附近  $N \times N$  大小的图像块,按照下面两个步骤生成随机树: 1) 把  $N \times N$  大小的图像块等距间隔的按照图 3(a) 的方法进行编码,则每个  $N \times N$  像素块分为  $Q$  个小块,每个小块中心像素  $P_i(i=0, 1, 2, \dots, Q)$  代表这个小块的值; 2) 由其中的任一  $P_i$  为基准,然后选取其他较分散的  $q$  个点作为比较点  $P_j$ (图 3(b)), 依次与  $P_i$  进行比较,分别根据  $P_j < P_i, P_j = P_i, P_j > P_i$  3 种情况进入随机树的左,中,右子节点,比较过程如图 4 所示。按照

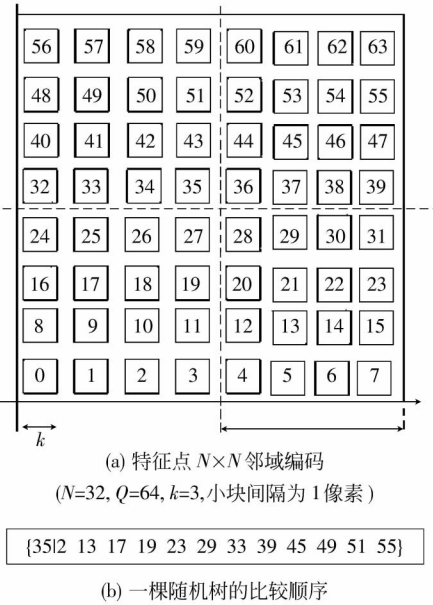


图 3 特征点图像块编码和随机树比较顺序

Fig. 3 Coding and sort sequence for Rdtree

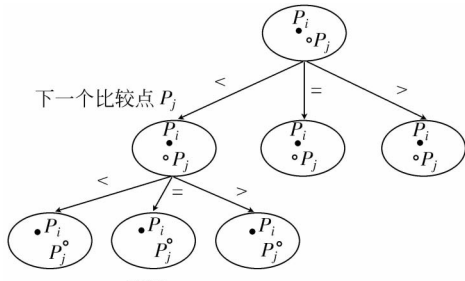


图 4 随机树的建立

Fig. 4 Rdtree construction

这样的方式,如果给定  $R$  个比较序列,则总共产生  $R$  棵随机树,每棵随机树为  $q$  层。

### 3 视频中特征点的检索、匹配、跟踪

#### 3.1 特征点检索、匹配及错匹配排除

视频中特征点的检索过程与特征点随机树的生成过程一致,即针对每个出现的 Harris 角点,按照其主梯度方向对齐到  $x$  轴,并采样对齐后像素点附近  $N \times N$  邻域的图像块,然后根据模板的  $R$  个比较序列,分别检索  $R$  棵随机树,得到该点在  $R$  棵随机树中命中次数最多的索引号  $s(s \in [0, T_0])$ ,如果  $s$  的命中次数大于  $R\lambda(\lambda \in (0, 1))$ ,则认为该像素点对应于模板的  $s$  点。

上面检索到的  $s$  点的可信度约为 60% ~ 80%<sup>[4]</sup>,需要进行错误匹配排除。文献[4]采用了 RANSAC<sup>[6]</sup> 进行误匹配排除。RANSAC 算法在初匹配小于 60% 的情况下比较耗时,且易产生误匹配,由此本文采用局部几何限制 (semi local constraint)<sup>[15-16]</sup> 结合单应矩阵迭代的方法进行错误匹配排除。过程如下: 半局部限制法(图 5)根据模板和视频两端同一初步匹配像素点最近的  $k$  个特征点为比较对象,如果这两个点集有超过一定数量  $k \times d(d$  是阈值)的点的标号相同,并且这些点的几何相对位置大概保持一致,则认为该点是一个初步匹配点。得到点集中的初步匹配点后,采用迭代的单应矩阵法去除不明显的错误匹配点。步骤为设视频中经过半局部限制初步匹配的点集记为  $V_0$ ,模板中对应于  $V_0$  的点集为  $H_0(H_0 \subset T_0)$ ,则存在一个单应矩阵  $M_0$ (式(1)  $i=0$  时),有  $V_0 = M_0 * H_0$ 。采用最小二乘法计算到  $M_0$ ,则得到对应

$$M_i = A_i x + b_i = R_\theta R_\phi^{-1} S R^\phi + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_2 \\ a_3 & a_4 & a_5 \end{bmatrix} \quad (1)$$

于  $V_0$  的  $V'_0(V'_0 = M_0 H_0)$ ,  $V'_0$  是估计值。

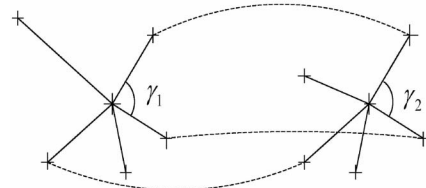


图 5 半局部限制

Fig. 5 Semi-local-constrain

从  $V_0$  中去掉  $\|V'_0 - V_0\| > c_0$  ( $c_0$  为常数) 的点, 记为  $V_1$ , 继续上一步操作, 模板中对应于  $V_1$  的点集  $H_1$  ( $H_1 \subset H_0 \subset T_0$ ), 由  $V_1 = M_1 H_1$  可求得  $M_1$  及  $V'_1$ , 由  $\|V'_1 - V_1\| > c_1$  去除误差大的点, 迭代进行, 当  $V'_{i+1} = V'_i = V_i$  或者  $i > K$  ( $K$  为常数) 时, 停止迭代, 得到较准确的匹配点集  $V_i, H_i$ 。在我们的实验中, 最后能保证至少 90% 的正确匹配。由于最小二乘迭代中偏差距离大的点被逐步排除, 最后保证剩下不到 10% 的错误匹配点在像素距离上误差不超过  $c_i$ , 确保了后面变换矩阵计算的精确性。图 6 给出了视频中两帧特征。点初步匹配及错误匹配排除后的比较结果。误匹配方法在初匹配可信度较低的情况下 (小于 60%), 时间效率比 RANSAC<sup>[6]</sup> 效率高。

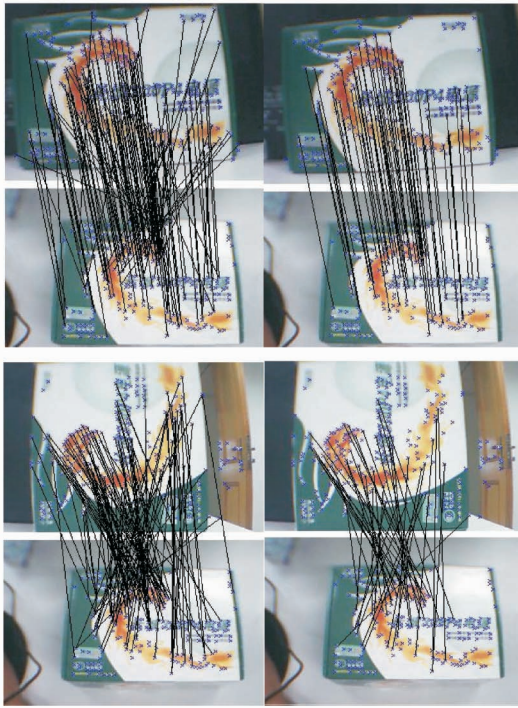


图 6 特征点初匹配(左)及错误匹配排除后(右)结果比较

Fig. 6 Initial points matching (left) and final matching after false matching discarded (right)

### 3.2 特征点的跟踪

检测到的特征点是预定义的跟踪点时, 程序开始对这些点进行跟踪。光流法<sup>[17]</sup>很好地描述了灰度图像在时间上的变化与场景中物体结构和运动的关系, 假定  $I(x, y, t)$  为视频中某时刻某像素的灰度图像值,  $I(x + v_x dt, y + v_y dt, t + dt)$  为下一时刻的该

点的灰度值, 则有  $I(x + v_x dt, y + v_y dt, t + dt) = I(x, y, t)$ , 进行泰勒展开得

$$I(x, y, t) + \frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} dt + O(dt^2) = I(x, y, t) \quad (2)$$

化简得到光流约束方程

$$\nabla I \cdot v_m + \frac{\partial I}{\partial t} = 0 \quad (3)$$

用 Lucas-Kanad 光流法<sup>[18]</sup>对特征点进行跟踪并采用  $N \times N$  大小图像块的模板匹配<sup>[19]</sup>方法进行跟踪错误校正。图 7 列出了视频中几帧特征点的跟踪效果图, 其中跟踪点集是动态变化的, 错误的点被去除, 新的检测到的跟踪点自动加入。

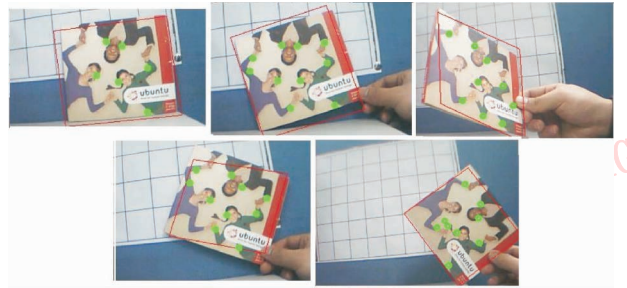


图 7 特征点的跟踪

Fig. 7 Tracking for feature points

### 3.3 变换矩阵计算

POSIT<sup>[10-11]</sup>给出了已知空间 4 点以及其在图像中的位置求取变换矩阵的算法。ARToolkit<sup>[3]</sup>给出了根据平面上构成矩形 4 点求取外部变换矩阵的算法。采用平面 POSIT<sup>[19]</sup>方法得到物体由世界坐标系到摄像机坐标系的旋转和平移矩阵。

## 4 算法流程

整个算法分为特征模板制作和视频处理两部分。

### 4.1 模板制作算法步骤

特征模板制作步骤如下:

- 1) 在视频中选择模板图像区域以及特征点集  $T_0, T_1, T_2$ , 见 2.1 节描述;
- 2) 生成不同参数的仿射和添加了高斯白噪声的图像, 并基于这些图像构造  $R$  棵  $q$  层随机检索树, 详细步骤见 2.2 节描述;
- 3) 保存这  $R$  棵随机树, 模板制作完毕。

## 4.2 视频处理端算法步骤

视频处理端算法包括特征点检测、跟踪、变换矩阵的计算几个方面,算法流程如下:

1) 对每帧输入的视频图像  $M$ , 检测 Harris 角点, 得到的角点集合记为  $C_0$ ;

2) 对  $C_0$  中的每一个角点  $i$ , 采用随机树法查找, 初步匹配到模板图像  $I$  中特征点  $j$ , 描述见 3.1;

3) 在  $M$  上取距  $i$  点距离最近的  $n$  个 Harris 角点, 构成集合  $X$ ; 在  $I$  上取距  $j$  点距离最近的  $n$  个 Harris 角点, 构成集合  $Y$ ; 对  $X, Y$  实施半几何限制, 若  $i$  是一个明显的误匹配点,  $i++$ , 回到步骤 2);

4) 集合  $C_0$  经过半几何限制后为  $C_1$ , 采用迭代的单应矩阵方法去除误差较大点后, 得到的点集记为  $C_2$ , 描述见 3.1;

5) 记前一帧的跟踪点集为  $G_0$ , 用 Lucas-Kanad 方法跟踪  $G_0$ , 得到对应  $G_0$  的当前帧点集  $G_1$ , 描述见 3.2 节;

6) 根据第 4) 步计算得到的单应矩阵排除  $G_1$  中的误差较大点, 得到集合  $G_2$ ;

7) 若集合  $G_2$  中点的个数大于 4, 则根据  $G_2$  中的点计算变换矩阵, 否则, 根据特征点检测集合  $C_2$  计算变换矩阵;

8) 根据计算得到的变换矩阵绘制虚拟物体, 本帧完毕, 到下帧。

## 4.3 算法时间效率分析

算法框架分为 2 个部分, 一是模板制作; 二是从视频检测。模板制作是一个离线过程, 不需要实时处理。视频检测主要分为特征点检测, 基于随机树的特征点初匹配, 基于半几何限制和单应矩阵的迭代误匹配消除, 光流法跟踪, 变换矩阵计算 5 个模块。其中特征点检测, 光流法跟踪以及变换矩阵计算这 3 个模块, 基本不会因为视频内容的改变而在时间花费上有很大的变化, 可认为是一个常量时间耗费。

基于随机树的特征点初匹配实际上是一个  $R$  棵  $q$  层 3 叉树的检索投票过程, 假定某帧频中检索得到  $m$  个角点, 则初匹配的时间复杂度为  $O(Rmq)$ 。

基于半几何限制和单应矩阵的迭代误匹配消除模块, 针对检测得到的初匹配点, 半几何限制和单应矩阵的迭代误匹配需要计算某点与附近点之间的几何关系, 假如某帧需要计算的点数为  $h$  个, 半几何

限制邻域点个数为  $g$ , 单应矩阵迭代的次数为  $l$ , 则时间复杂度为:  $O(hgl)$ 。

## 5 实验及分析

### 5.1 实验结果

采用 VisualStudio2003 和 OpenGL 开发了整个算法框架, 操作界面如图 8 所示, 界面左下角区域为模板图像, 左上角区域为来自于摄像机的实时视频序列, 它们之间的连线表示模板和实时视频流中特征点和跟踪点的匹配情况。右上角区域为弹出的 OpenGL 窗口, 用于真实视频图像序列和虚拟 3D 物体的实时融合绘制。相关参数设置如下: 2.1 节中提到的参数取  $k=2, \alpha=200, \beta=\max(\lambda_1, \lambda_2)/5$ , 2.2 节的参数  $N=32, Q=64, k=3, R=20, q=11$ 。3.1 节的参数中, 取  $\chi=0.6, k=5, d=0.8, K=20, c_i$  从最大值 20 渐变到最小值 5。4.3 节的参数中,  $m$  大致为 200~300,  $h$  约为 50,  $g$  约为 10,  $l$  取 10 到 20 次, 实验中, 这些参数在针对不同图像模板特征点检测和跟踪时, 基本没有变化。实验中采用的 AnyvisionWeb Camera,  $f_x=160.275$  mm,  $f_y=172.279$  mm,  $d_x$  与  $d_y$  分别约为 0.421875 (mm/pixel) 和 0.445833 (mm/pixel)。程序运行在主频 2.5, 内存 512M 的 PC 上, 每秒 16 帧左右, 其中特征点检测, 光流法跟踪以及变换矩阵计算这 3 个模块每帧耗时大约 30 ms, 基于随机树的特征点初匹配和半几何限制结合单应矩阵的迭代误匹配消除模块分别耗时 15 ms 和 8 ms 左右。图 9 列出了程序运行中的几帧视频截图, (a)(b) 中左下角图片为模板, 连线表示点匹配, 右上角图片为虚实融合结果 (模板大小为 140 mm × 120 mm, 绘制的虚拟茶壶和立方体的大小分别为 80 mm 和 60 mm)。(c)(d)(e) 为实验过程中另外的一些虚实融合视频截图, 其中另外一个模板的大小为 210 mm × 170 mm。当模板被部分遮挡或不可见时, 由于特征点检测不足, 如 (d) 中第 2 幅图所示, 绘制虚拟物体产生少许飘移, 总体来讲, 虚实融合的视觉效果是稳定的。

### 5.2 分析及未来工作

算法的一个特色之处在于从视频中检索到预定义的特征点后, 采用 Lucas-Kanad 光流法对这些特征点进行跟踪, 达到保持视觉稳定性的目的。特别

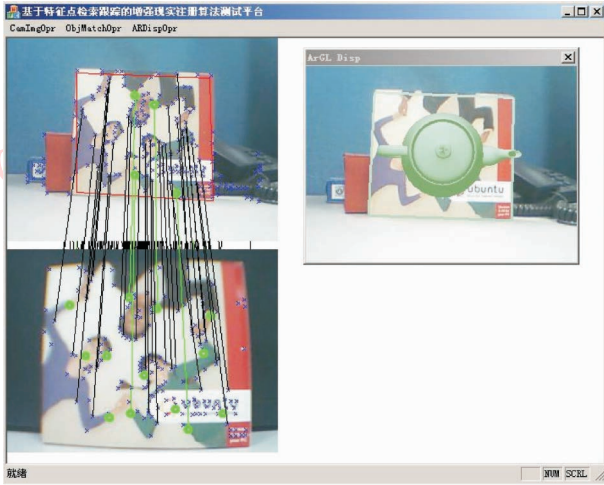


图 8 实时视频增强现实系统界面

Fig. 8 Interface for real-time video augmented system

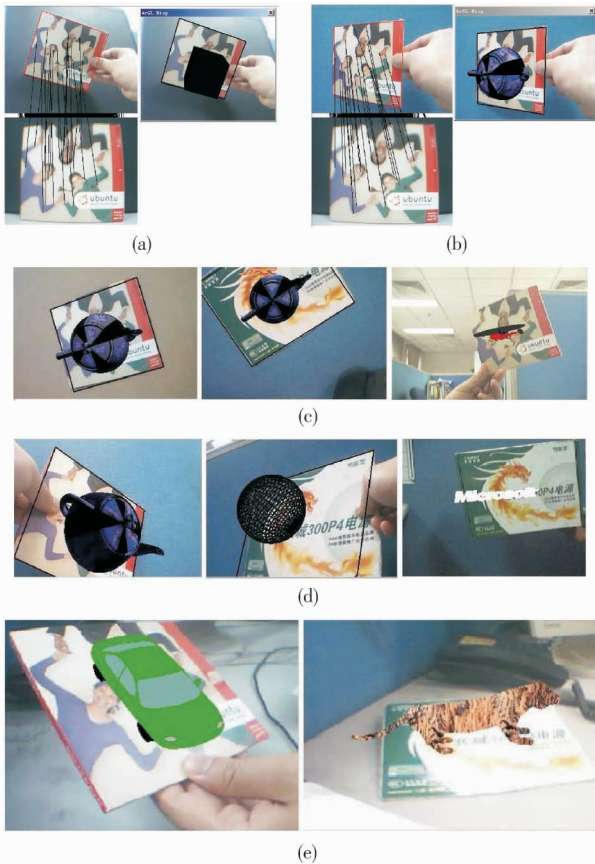


图 9 虚实融合结果

Fig. 9 Augmented results

是在摄像头静止的时候,基本保证虚拟物体的稳定。图 10 列出了分别采用 Lucas-Kanad 光流法对预定义特征点进行跟踪和不采用光流法直接根据 RdTree 检索特征点两种方法,对同一段静止视频和

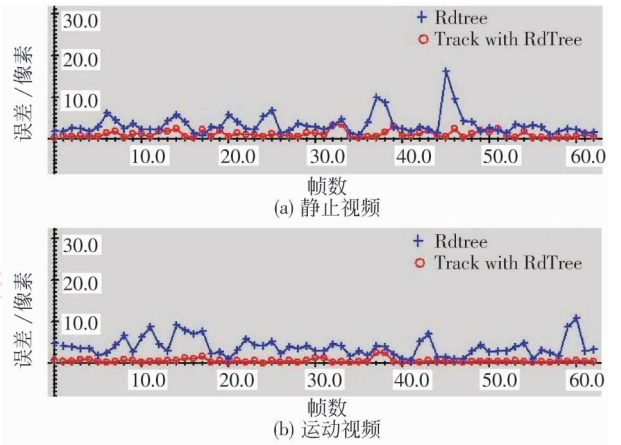


图 10 两种方法偏移误差比较

Fig. 10 Compare our method with Rdtree retrieving

非静止视频得到的特征点与真实值的偏移距离均方差,绘制的 2 条曲线示意图。可以看出,由于特征点跟踪方法的引入,较好的保证了虚拟物体的不抖动。

从应用角度看,因为是基于自然特征点的注册算法,图像模板可以是任何一幅自然图片,减少了场景的人工雕饰,为小场景增强现实真实化和自然化提供了途径。

误差方面,采用特征点投影在图像中的估计值和真实观测得到的像素位值的差值均值作为评价标准,

$$d = \frac{\sum_{i=1}^n \sqrt{(X_i^1 - X_i^0)^2}}{n} \quad (4)$$

式中,各特征点投影在图像中的估计值为  $X_i^1$  ( $X_i^1$  为 2 维向量,  $i = 1, 2, \dots, n$ ), 各特征点在视频图像中实际观测像素值为  $X_i^0$ 。实验中,误差  $d$  大约在 10 像素以内。分析可知,误差主要来源于 2 个方面:一是图像处理过程中,部分误匹配点没有完全去除导致的变换矩阵不精确引起的反投影误差;另一方面是算法假定摄像机投影中心在图像中心所引起的偏移误差,下一步将考虑在上面 2 个方面改进算法。

其他方面,在普通 PC 上,本算法运行速度(16 帧/s)距离完全的实时要求(24 帧/s)还有一定的差距,期望能通过减少特征点检索来提高速度。另外,在摄像机运动剧变或者光照突变的情况下,由于光流法本身的特性,跟踪点不稳定导致虚拟物体还是会抖动,拟进一步采用扩展卡尔曼滤波或者粒子滤波的方法来平滑这种抖动。

## 参考文献 (References)

- [ 1 ] Azuma R, Baillet Y, Behringer R, et al. Recent advances in augmented reality [ J ]. IEEE Computer Graphics and Application, 2001, 21(6) : 34-47.
- [ 2 ] Azuma, Ronald T. A survey of augmented reality [ J ]. Teleoperators and Virtual Environments, 1997, 6(4) :355-385.
- [ 3 ] Kato H, Billinghurst M. Marker tracking and HMD calibration for a video-based augmented reality conferencing system [ C ]// Proceedings of 2nd IEEE and ACM International Workshop on Augmented Reality. Washington, DC, USA: IEEE Computer Society, 1999: 85-94.
- [ 4 ] Lepetit V, Lagger P, Fua P. Randomized trees for real-time keypoint recognition [ C ]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC, USA: IEEE Computer Society, 2005: 775-781.
- [ 5 ] Amit Y, Geman D. Shape quantization and recognition with randomized trees [ J ]. Neural Computation, 1997, 9(7) : 1545-1588.
- [ 6 ] Fischler M A, Bolles R C. Random sample consensus; a paradigm for model fitting with application to image analysis and automated cartography [ J ]. Communication Association and Computing Machine, 1981, 24(6) :381-395.
- [ 7 ] Gilles Simon, Andrew W Fitzgibbon, Andrew Zisserman. Markerless tracking using planar structures in the scene [ C ]// Proceedings of IEEE and ACM International Symposium on Augmented Reality. Washington, DC, USA: IEEE Computer Society, 2000: 120-128.
- [ 8 ] Andrew I, Eric M, Francois C. A real-time tracker for markerless augmented reality [ C ]// Proceedings of IEEE / ACM International Symposium on Mixed and Augmented Reality. Washington, DC, USA: IEEE Computer Society, 2003: 36-45.
- [ 9 ] Muriel Pressigout, Andrew I Comport, Eric Marchand, et al. Real-time markerless tracking for augmented reality: the virtual visual servoing framework [ J ]. IEEE Transactions on Visualization and Computer Graphics Archive, 2006, 12(4) : 615 - 628.
- [ 10 ] DeMenthon D, Davis L S. Model-based object pose in 25 lines of code [ J ]. International Journal of Computer Vision, 1995, 15(1-2) : 123-141
- [ 11 ] Oberkampf, DeMenthon D, Davis L S. Iterative pose estimation using coplanar points [ J ]. Computer Vision and Image Understanding, 1996, 63(3) : 495-511.
- [ 12 ] Harris C, Stephens M. A combined corner and edge detector [ C ]// Proceedings of the 4th Alvey Vision Conference, Berlin, Germany: Springer-Verlag, 1988: 147-151.
- [ 13 ] Shi J, Tomasi C. Good features to track [ C ]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC, USA: IEEE Computer Society, 1994: 593-600.
- [ 14 ] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints [ J ]. International Journal of Computer Vision, 2004, 20(2) : 91-110.
- [ 15 ] Schmid C, Mohr R. Local grayvalue invariants for image retrieval [ J ]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(5) : 530-534.
- [ 16 ] Zhang Z, Deriche R, Faugeras O, et al. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry [ J ]. Artificial Intelligence, 1995, 78(1/2) : 87-119.
- [ 17 ] Ma Songde, Zhang Zhengyou. Computer Vision: Fundamentals of Computational Theory and Algorithms [ M ]. Beijing: Science Press, 1998. [ 马颂德, 张正友著. 计算机视觉: 计算理论与算法基础 [ M ]. 北京: 科学出版社, 1998. ]
- [ 18 ] Bouguet J Y. Pyramidal implementation of the lucas kanade feature tracker [ R ]. Intel Corporation, Microprocessor Research Labs, 2000.
- [ 19 ] Lewis J P. Fast template matching [ C ]// Vision Interface. Quebec City, Canada: Canadian Image Processing and Pattern Recognition Society, 1995: 120-123.