

# 基于 Android 平台的增强现实算法<sup>①</sup>

阮文惠, 薛亚娣

(兰州文理学院, 兰州 730000)

**摘要:** 为了获得更加理想的增强现实效果, 提出一种基于利用 Android 平台的增强现实算法. 首先采用 FAST 算法确定两幅图像中的特征角点, 并采用 SURF 生成特征点描述符, 然后采用快速近似邻近点搜索进行图像匹配, 最后在 Android 平台上实现算法, 并采用仿真实验测试算法的性能. 仿真结果表明, 本文算法结合 Android 的优点, 可以较好的满足增强现实的实时性的要求, 而且具有较好的鲁棒性, 克服了传统增强现实技术的局限性.

**关键词:** 增强现实; SURF 特征; 快速匹配; 注册算法

## Augmented Reality Registration Algorithm Based on Android System

RUAN Wen-Hui, XUE Ya-Di

(Lanzhou University of Arts and Science, Lanzhou 730000, China)

**Abstract:** In order to obtain more ideal effect of the augmented reality, this paper puts forward an augmented reality algorithm based on Android platform. Firstly, the feature positions in the two image corner points are determined by using FAST algorithm, and SURF is used to generate feature descriptor, and then the fast approximate proximal point search algorithm is used to match, finally, implemented in the Android platform and the performance is test by simulation experiment. The simulation results show that, the proposed algorithm has the advantages of Android and can meet the augmented reality real-time requirements, but also has good robustness, overcomes the limitation of traditional augmented reality technology.

**Key words:** augmented reality; SURF feature; fast match; registration algorithm

## 1 引言

随着经济的不断发展, 信息化对人类社会产生了深刻影响, 同时随着计算机技术的不断成熟, 智能手机日益得到普及, Android 是 Google 开发的一套基于 Linux 的开源操作系统, 其具有可移植性强、高稳定性、可靠性、集成度高等优点, 可以针对不同的应用和硬件环境进行调整, 在智能手机和平板电脑中得到了广泛的应用<sup>[1]</sup>.

增强现实(Augmented Reality, AR)是一种利用计算机产生的虚拟信息对使用者所看到的真实世界景象进行增强或扩张的技术, 实际上就是将计算机生成的虚拟物体或场景叠加到真实场景中, 从而实现对现实的“增强”<sup>[2]</sup>. 当前增强实现主要包括人工标识和无标

识两种, 人工方式对光照十分敏感, 应用范围受限, 因此无标识增强现实技术成为当前的主要研究方向<sup>[3]</sup>. 在无标识增强现实技术中, 图像特征点的提取与匹配是获得更好的三维跟踪注册效果的关键<sup>[4]</sup>. Harris 算法以其计算简单, 提取角点分布均匀稳定和对图像的平移、旋转以及噪声干扰具有一定的鲁棒性等特点被广泛使用. 但由于 Harris 算法在计算角点响应函数时采用了差分方向导数的方式, 与理想的方向导数之间有误差从而使角点定位有偏差, 难以满足 AR 系统对匹配精度的要求<sup>[5]</sup>; SIFT(Scale Invariant Feature Transform)算法基于尺度空间和使用邻域梯度信息对特征进行提取和描述, 使得 SIFT 特征对图像的旋转、缩放和噪声干扰甚至仿射变换都具有较好的鲁棒性.

<sup>①</sup> 基金项目: 甘肃教育厅研究生导师项目(1215-04)

收稿时间: 2014-09-10; 收到修改稿时间: 2014-11-02

但其存在计算量大的缺点,所以无法满足 AR 系统高实时性的要求,且 SIFT 算法提取的特征点大部分不是边缘点和角点,不能很好地反映图像的结构特征,不利于 AR 系统对目标的识别及定位<sup>[6]</sup>;SURF(Speed Up Robust Features)算子,采用积分图像和箱式滤波器在保证特征点提取与匹配精度的同时,在一定程度上提高了处理速度<sup>[7]</sup>;ORB(Oriented Brief)算法是一种高效的二进制局部特征描述符,但是 ORB 在算法上并没有做一个严密的流程设计,并没有解决尺度不变性<sup>[8]</sup>. FAST 算法在角点探测中速度较快,但其旋转不变性差,光照不变性差,鲁棒性差;SURF 其运算速度无法满足增强现实系统的实时性要求,而它却能很好地满足基于自然特征点跟踪这一需要.所以选择 FAST 算法和 SURF 算法相结合.

为了获得了更加理想的增强现实效果,基于 Android 平台的增强现实算法,并采用仿真实验测试算法的性能,仿真结果表明,本文算法改善了增强现实的速度,可以很好的满足增强现实的实时性的要求,而且对光照、遮挡不敏感,鲁棒性强,具有一定的实际应用价值.

## 2 增强现实系统

### 2.1 Android 系统框架

Android 能够作为目前手机市场占有率最大的智能终端,主要得益于其开源的优势,开放源代码使 Android 在应用程序的开发上具备显著的优势,基于 Linux 的 Android 可以与各种基于 Linux 平台的 ARM 嵌入式处理器进行通信.从架构图看,Android 分为四层,分别是应用程序层、应用程序框架层、系统运行库层和 Linux 核心层<sup>[9]</sup>.应用开发人员大部分情况下是和系统运行库层打交道.应用程序架构设计简化了组件的重要性,任一应用程序都可以发布它的功能块,并供其他的任何应用程序使用,使用户可以方便的替换程序组件.系统运行库包括系统库和运行库,系统库中包含一些 C/C++库,这些库能被 Android 系统中不同的组件使用.核心库提供了 JAVA 编程语言核心库的大多数功能,其中最重要的是缓存器型态的 Dalvik 虚拟机,可以使用最少的内存资源来执行,同时执行多个虚拟机特性,运行的速度较快,且硬件实现比较容易,具体如图所示.

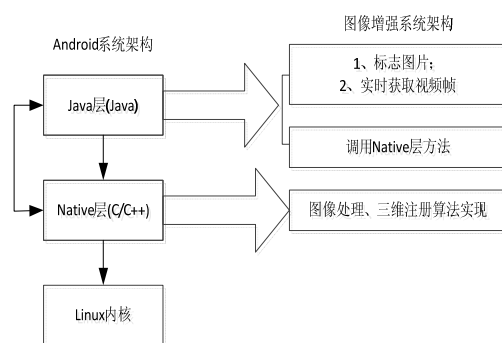


图 1 Android 系统框架

### 2.2 系统详细设计

首先在 Java 层从本地获取模板图片并通过摄像头实时获取视频帧,将数据通过 NDK 传输到 Native 层,使用 SURF 算法对图像的特征点进行提取和描述,然后基于欧式对描述子集合进行匹配,并对删除不匹配点理,最后进行三维模型的注册,实现虚实融合的效果,具体流程如图 2 所示.

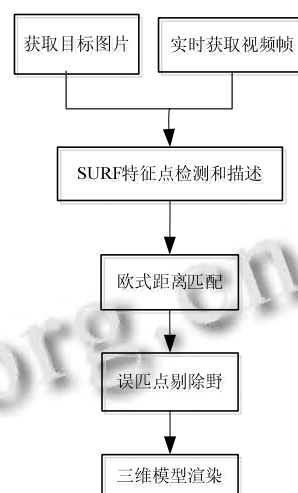


图 2 本系统的工作流程

## 3 增强现实的相关技术

### 3.1 特征点提取

FAST 特征提取的角点响应函数定义为:

$$N = \sum_{x \in circle(p)} |I(x) - I(p)| < \epsilon_d \quad (1)$$

式中,  $I(x)$  为圆周上任意一点的灰度,  $I(p)$  为圆心的灰度,  $\epsilon_d$  为灰度值差得阈值<sup>[10]</sup>.

候选点周围的圆的选取半径是一个很重要的参数,为了简单高效,采用半径为 3,共有 16 个周边像素需要比较.通常只使用 N 个周边像素来比较,广泛使用

的是 FAST-12. 具体情况如图 3 所示, 此图给出了以该点为中心的圆形区域的模板情况, 圆形区域为一个半径为 3 个像元的离散化的区域, 像元  $p$  为候选角点, 其中高亮度像元为角点检测中使用到的像元, 最外围的像素点按顺时针顺序依次编号为 1~16. 利用 FAST 检测到的特征点的如图 4 所示.

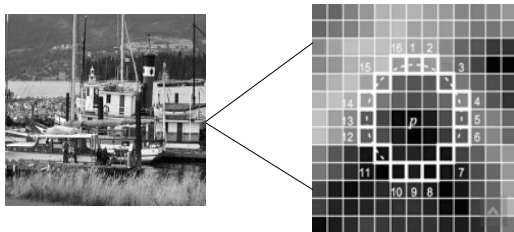


图 3 FAST 算法检测的模板示意图

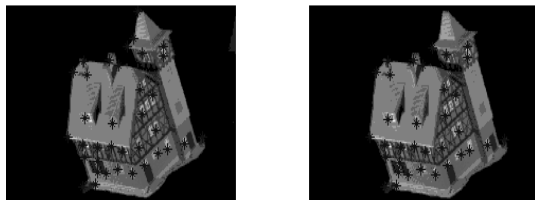


图 4 利用 FAST 检测到的特征点

### 3.2 特征点的描述

使用 SURF 选取特征点的主方向. 首先统计特征点领域内的 Haar 小波特征, 将以获取的稳定特征点为中心, 计算以  $6\sigma$  为半径的圆形邻域内所有的点在  $z, y$  方向上的 Haar 小波响应值, 根据每个点到特征点的距离不同分配给点不同的高斯权重, 越远权重值越大, 距离越近权重值越小. 在特征点的领域内, 统计在滑动方向上覆盖  $\pi/3$  的区域内扇形内所有点的水平 Haar 小波特征和垂直 Haar 小波特征的矢量总和, 这样一个扇形得到了一个值<sup>[11]</sup>. 然后  $\pi/3$  度扇形以一定间隔进行旋转, 最后将最大值那个扇形的方向作为该特征点的主方向. 接下来构造 SURF 特征点描述算子, 沿着前一步选定的主方向在特征点周围取一个正方形框, 边长为  $20\sigma$ . 然后将该区域规律地分为  $4 \times 4$  的方形子区域, 在每个子区域统计  $5 \times 5\sigma$  范围内像素的相对于主方向的小波响应值, 即水平方向  $dx$  和垂直方向  $dy$  的 haar 小波响应值, 并将高斯权值赋给该响应值. 该 haar 小波特征为水平方向值之和, 水平方向绝对值之和, 垂直方向之和, 垂直方向绝对值之和. 这样每一个子区域就得到一个四维的向量  $V_{sub}=(\sum dx, \sum |dx|,$

$\sum dy, \sum |dy|)$ , 对于每一个特征点形成了  $4 \times (4 \times 4)$  维度的描述向量, 然后进行向量的归一化, 增加其对几何变换的鲁棒性, 减小局部误差. 具体如图 5 所示.

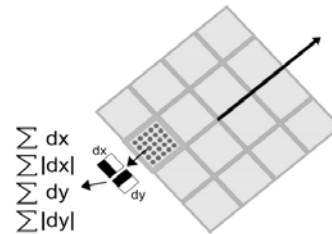


图 5 SURF 特征点描述算子产生过程

### 3.3 特征的匹配

为提高匹配的正确率, 首先采用快速近似邻近点搜索算法找到具有最近距离的匹配点对, 定义如式(2)所示.

$$D = \sqrt{(x_1 - x'_1)^2 + (x_2 - x'_2)^2 + \dots + (x_{64} - x'_{64})^2} \quad (2)$$

式中  $(x_1, x_2, \dots, x_{64}), (x'_1, x'_2, \dots, x'_{64})$  为待匹配的两个特征点的 SURF 特征向量.

### 3.4 误匹配点的剔除

经过上面的特征匹配后得到的初匹配点集中还存在一定数量的误匹配点, 若不进行剔除将会导致单应性矩阵和摄像机内外参数求解上的误差从而影响注册精度, 因此要对初匹配点集进行误匹配点剔除, 文采用了统计特征点间距离均值误差的方法剔除误匹配点.

设  $A_0, B_0$  匹配点, 首先在待匹配图像中的初匹配点集里选取一定数量的点  $A_i (i=1, 2, \dots, n)$  并计算  $A_0$  与  $A_i$  描述符间的欧式距离  $D_i$ , 再在参考图像中选择与之对应的初匹配点  $B_i (i=1, 2, \dots, n)$  并计算  $B_0$  与  $B_i$  描述符间的欧式距离  $d_i$ , 然后计算  $D_i/d_i$ , 记为  $RD_i$ ,  $RD_i$  应分布在平均值附近, 因此构造响应函数:

$$num = \sum_{i=1}^n V_i \quad (3)$$

若  $num < n/2$  则认为  $A_0, B_0$  是一对正确匹配, 否则剔除这对匹配点. 阈值  $T$  的选取主要根据  $d_i$ , 当  $d_i$  较小时小的匹配误差会产生较大响应, 阈值  $T$  应取大; 当  $d_i$  较大时匹配误差的响应较小, 阈值  $T$  应取小. 以上方法在视角变换不大的场合有较好效果, 在满足 AR 系统要求的前提下降低了运算量.

## 4 仿真实验

### 4.1 仿真环境

为了验证系统的性能, 在 Pentium 双核 2.85GHZ, 4G RAM, Windows XP 操作系统, 采用 VC++ 编程实现仿真实验.

### 4.2 结果与分析

#### 4.2.1 匹配性能的对比

为了测试本文匹配算法的性能, 采用文献[12]、[13]算法进行对比实验, 匹配点对、匹配正确率以及匹配时间(ms)如表 1 所示. 从表 1 可以看出, 相对于对比算法, 本文算法匹配的点对数有所减少, 去除了一些伪匹配, 弥补了之前损失的一定时间效率, 匹配正确率得到了提高, 提高了算法的稳定性, 且保证了算法的实时性.

表 1 与其它系统的匹配性能对比

名称	匹配对数量	匹配正确率(%)	匹配时间(ms)
本文系统	180	95	500
文献[12]	280	92	651
文献[13]	224	83	620

#### 4.2.2 理想环境下的匹配结果对比

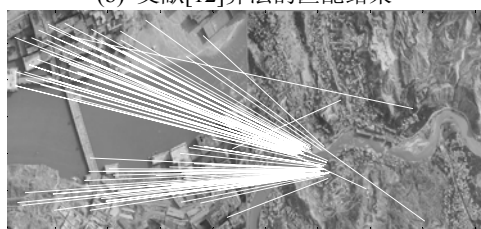
对于图 6(a)的图像, 它们的匹配效果如如图 6(b)~(d)所示. 从 6 可知, 相对于对比算法, 本文算法的匹配效果更优, 对比结果证明了本文算法有优越性.



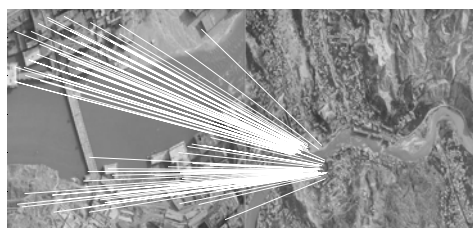
(a) 原图像



(b) 文献[12]算法的匹配结果



(c) 文献[13]算法的匹配结果

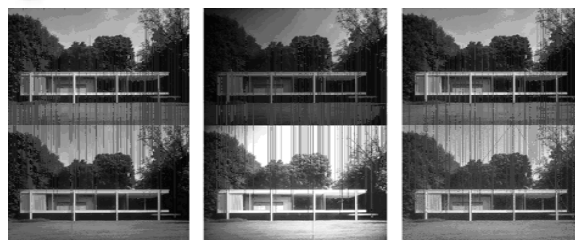


(b) 本文算法的匹配结果

图 6 不同算法的匹配结果对比

#### 4.2.3 非理想环境下的匹配结果

为了测试本文算法的鲁棒性, 对光照变化、高期噪声图像进行仿真, 验证算法的鲁棒性和稳定性, 仿真如图 7 所示. 从图 7 可知, 本算法在 Android 系统下可以有很好的扩展性以及稳定性.



(a) 原图

(b) 光照变化

(c) 高斯噪声

图 7 非理想环境的匹配结果

#### 4.2.4 基于 Android 的虚实注册效果

基于 Android 平台的增强现实算法的虚拟现实注册效果如图 8 所示, 从图 8 可知, 本文系统可以较好的实现虚拟现实增强, 获得比较理想的结果.



图 8 基于 Android 系统下的增强实现效果

## 5 结 语

为了获得更加理想的增强现实效果, 提出一种基于 Android 平台的增强现实算法. 首先用 FAST 算法确定两幅图像中的特征角点, 然后采用 SURF 生成特征点描述符, 然后采用快速近似邻近点搜索进行图像匹配, 最后在 Android 平台上实现算法. 仿真结果表明, 该算法能够在 Android 上实现增强现实的应用, 具有较强的鲁棒性, 可以能够满足实时性的要求, 具有一定实际应用价值.

### 参 考 文 献

- 1 朱淼良, 姚远, 蒋云良, 等. 增强现实综述. 中国图象图形学报, 2004, 9(7): 767-774.
- 2 Setohara H, Kato H, Kawamoto K, et al. A simple solution of occlusion problem in augmented reality and its application for interaction. Trans. of the Virtual Reality Society of Japan, 2013, 9(4): 387-395.
- 3 Tian Y, Guan T, Wang C. Real-time occlusion handling in augmented reality based on an object tracking approach. IEEE Computer Graphics and Applications, 2010, 10(4): 2885-2900.
- 4 Tian Y, Guan T, Wang C. Real time occlusion handling in augmented reality based on an object tracking approach. Sensors, 2010, 10(4): 2885-2900.
- 5 管涛, 李利军, 段利亚, 等. 基于全局单应性变换的虚实注册方法. 华中科技大学学报(自然科学版), 2007, 35(4): 100-102.
- 6 李聪. 基于 FPGA 的增强现实人工标识识别. 电子科技, 2014, 27(5): 119-123.
- 7 施琦, 王涌天, 陈靖. 一种基于视觉的增强现实三维注册算法. 中国图象图形学报, 2002, 7(7): 679-68.
- 8 张钰, 陈靖, 王涌天, 周琪. 增强现实浏览器的密集热点定位与显示. 计算机应用, 2014, 34(5): 1435-1438, 1462.
- 9 陈岭, 季诚夏, 黄伟恺, 等. 面向移动增强现实的屏幕外对象可视化方法. 计算机辅助设计与图形学学报, 2012, 24(6): 832-838.
- 10 曾峦, 王元钦, 谭久彬. 改进的 SIFT 特征提取和匹配算法. 光学精密工程, 2011, 19(6): 1391-1397.
- 11 赵钦君, 赵东标. Harris-SIFT 算法及其在双目立体视觉中的应用. 电子科技大学学报, 2010, 39(4): 546-550.
- 12 Wagner D, Schmalstieg D, Graz. Making augmented reality practical on mobile phones. IEEE Computer Graphics and Applications, 2009, 9(4): 6-9.
- 13 Henderson S, Feiner S. Opportunistic tangible user interfaces for augmented reality. IEEE Trans. on Visualization and Computer Graphics, 2010, 16(1): 4-16.