

基于 ARToolKit 的增强现实系统开发与应用

Exploiting and Applying of Augmented Reality System Based on ARToolKit

任波 管涛 李利军 (华中科技大学数字化工程与仿真中心 武汉 430074)
段利亚 (青岛科技大学自动化与电子工程学院 266042)

摘要:介绍了增强现实系统二次开发工具 ARToolKit,给出了其体系结构及工作流程,详细讨论了利用 ARToolKit 开发增强现实应用系统的流程。并给出了一个实例—基于增强现实技术的虚拟样板房展示系统。

关键词:增强现实 ARToolKit 三维注册

1 引言

增强现实 (Augmented Reality 简称 AR) 是虚拟现实技术的一个重要分支,也是近年来一个研究热点。增强现实与虚拟现实 (Virtual Reality, 简称 VR) 不同,VR 技术让用户完全沉浸在由计算机生成的虚拟环境中,而 AR 的最终目的则是把计算机产生的虚拟场景无缝的融合到用户所能观察到的真实世界当中。使用 AR 技术可以增强用户的视觉感受,延伸用户视觉系统的功能,并使其能以一种更加自然的方式与真实和虚拟相融合的环境进行交互,更好地完成某些任务。增强现实系统作为一种新型的媒介,已经逐渐应用到从医学到军事等各个领域^[1]。

开发增强现实系统的难点在于如何实时、精确的计算观察者视点相对于真实世界的位置和姿态,使得虚拟场景能够与真实世界无缝融合(即三维注册)。当前流行的基于计算机视觉的三维注册方法主要有两种:基于标识的三维注册和基于自然特征的三维注册。国内外研究人员在以上两方面开展了大量的研究工作,提出了一些解决方法,然而这些工作大部分都处于实验室研究阶段,真正形成产品的较少。在基于自然特征的三维注册技术没有发展成熟之前,基于标识的三维注册将是当前增强现实应用系统中的主要注册方法。

该文介绍了一个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系统二次开发工具—ARToolKit^[2]。在介绍其体系结构及工作流程的基础上,详细讨论了基于 ARToolKit 的增强现实应用系统的开

发流程,并给出了一个实例—基于增强现实技术的虚拟样板房展示系统。

2 ARToolKit 简介

ARToolKit 是一套基于 C 语言的增强现实系统二次开发包,能够在 SGI IRIX, PC Linux 以及 PC Windows 95/98/NT/2000 等不同的操作系统平台上运行。它利用计算机视觉技术来计算观察者视点相对于已知标识的位置和姿态,并且同时支持基于视觉或视频的增强现实应用。其实时、精确的三维注册功能使得工程人员能够非常方便、快捷的开发增强现实应用系统。

2.1 体系结构

ARToolKit 主要由图 1 所示的几个函数库组成,分别是:

AR32.lib: 包括摄像机校正与参数收集、目标识别与跟踪模块。主要完成摄像机定标,标识识别与三维注册等功能。

ARvideoWin32.lib: 基于微软视频开发包 MS Vision SDK 的视频处理函数库。主要完成图像实时采集功能。

ARgsub32.lib: 基于 OpenGL 的图形处理函数库。完成图像的实时显示,三维虚拟场景的实时渲染等功能。

以上几个模块中除 ARvideoWin32.lib 外,其它部分的源代码都是对用户开放的,开发人员可以根据需要对其进行修改和补充。

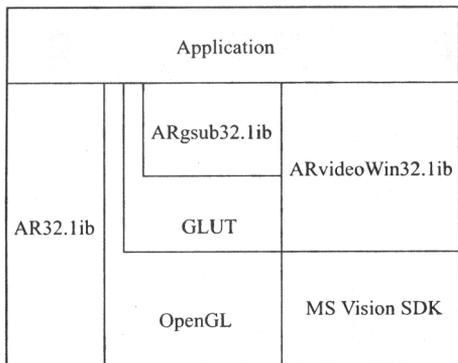


图 1 ARToolkit 体系结构

2.2 ARTOOLKIT 坐标系

如何有效的计算世界坐标系与摄像机坐标系之间的转换矩阵,是增强现实系统所面临的主要问题之一,ARToolKit 也不例外。ARToolKIT 坐标系如图 2 所示。由于采用了基于标识的虚实注册方法,ARToolKit 中的注册问题就转变成标识坐标系与摄像机坐标系之间变换矩阵的求解。两者之间的变换关系如式 1 所示,其中为待求变换矩阵。文献[3]中详细介绍了的求解过程,不再赘述。

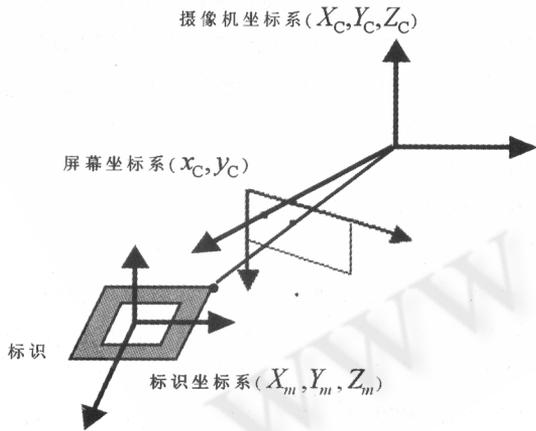


图 2 ARTOOLKIT 坐标系

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} & V_{13} & W_x \\ V_{21} & V_{22} & V_{23} & W_y \\ V_{31} & V_{32} & V_{33} & W_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} =$$

$$\begin{bmatrix} V_{3 \times 3} & W_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = T_{cm} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (1)$$

2.3 工作流程

ARToolKit 利用计算机视觉技术来计算摄像机相对于标识的距离与位置,工作流程如图 3 所示,首先根据用户设定的阈值将采集到的一帧彩色图像转换成一幅二值(黑白)图像,然后对该二值图像进行连通域分析,找出其中所有的四边形区域作为候选匹配区域,将每一候选区域与模板库中的模板进行匹配,如果产生匹配,则 ARToolkit 认为找到了一个标识,利用该标识区域的变形来计算摄像机相对于已知标识的位置和姿态,最后根据得到的变换矩阵实现虚实之间的注册。

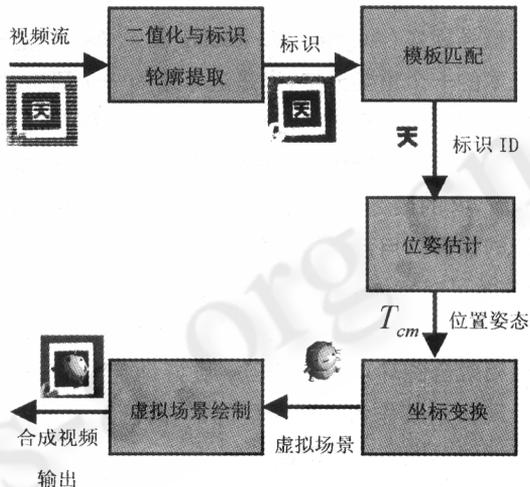


图 3 ARToolkit 工作流程

3 基于 ARToolkit 的增强现实系统开发

利用 ARToolkit 开发增强现实系统之前必须进行摄像头定标、模板制作与训练两项工作。

3.1 摄像机定标

这里的定标指的是摄像头内部几何参数的确定,主要包括:焦距、像面中心、像素的水平和垂直间距 Δ 、 Δ 、倾斜率等。系统运行时正是根据这些内部参数来实时计算摄像头与目标标识之间的三维变换矩阵,因此这些内部参数的准确与否将直接影响到系统的注

册精度。ARToolKit 提供了用于摄像头定标的例程,利用它可以非常方便的确定摄像头的内部参数。

3.2 模板制作与训练

在利用 ARToolKit 开发增强现实系统之前,必须制作标识,系统正是根据这些不同的标识返回的不同 ID 值来叠加不同的虚拟场景。ARToolKit 提供的标准模板是一个带黑色边框的正方形,开发人员可以根据需要向白色区域中添加自定义图形,来制作不同的标识,图 4 所示为系统自带的几个标识。



图 4 ARToolKit 标识

在使用标识之前必须进行训练,使系统根据不同的标识生成对应的模板文件。ARToolKit 提供了一个函数 `arSavePatt` 来完成上述工作。

制作标识时须注意标识之间的区分度,应使标识之间的区别尽量大,以免系统识别时发生混淆。

3.3 开发流程

利用 ARToolKit 进行系统开发必须经过如下几个步骤:

- (1) 初始化摄像机,读取摄像机参数和标准模板库。
- (2) 获取一帧图像。
- (3) 标识识别。
- (4) 视点与目标标识之间转换矩阵的计算。
- (5) 根据转换矩阵叠加虚拟场景。
- (6) 关闭视频流。

其中第二步到第五步是一个循环过程,直到程序结束。第一步和第六步只在系统初始化和退出时执行。

以上几步的程序实现如下

主函数

```
main(int argc, char * argv[ ])
```

```
{
    init(); //系统初始化
    arVideoCapStart(); //开启视频流
```

```
    argMainLoop( NULL, keyEvent, mainLoop ); //主
    循环
}
```

函数 `init` 实现如下

```
static void init( void )
{
    ARParam wparam;
    /* 初始化视频采集参数 */
    if( arVideoOpen( vconf ) < 0 ) exit(0);
    if( arParamLoad( cparam_name, 1, &wparam ) <
        0 ) exit(0);
    arParamChangeSize( &wparam, xsize, ysize,
        &cparam );
    arInitCparam( &cparam );
    /* 读取模板文件 */
    if( ( patt_id = arLoadPatt( patt_name ) ) < 0 )
    {
        exit(0);
    }
    argInit( &cparam, 2.0, 0, 0, 0, 0 );
}
```

函数 `mainLoop` 实现如下

```
static void mainLoop( void
){
    /* 相关变量定义 */
    ARUint8 * dataPtr;
    ARMarkerInfo * marker_info;
    int marker_num;
    int j, k;
    /* 获取一帧图像 */
    dataPtr = ( ARUint8 * ) arVideoGetImage();
    argDisplmage( dataPtr, 0, 0 );
    /* 连通域分析 */
    if( arDetectMarker( dataPtr, thresh, &marker_info,
        &marker_num ) < 0 )
    {
        exit(0);
    }
    arVideoCapNext();
    k = -1;
```

```

/* 标识匹配 */
for( j = 0; j < marker_num; j + + )
{
    if( patt_id == marker_info[j].id ) {
        if( k == -1 ) k = j;
        else if( marker_info[k].cf < marker_info
[j].cf ) k = j;
    }
}
if( k == -1 ) {
    argSwapBuffers();
    return;
}
/* 变换矩阵计算 */
arGetTransMat( &marker_info[k], patt_center,
patt_width, patt_trans);
/* 叠加虚拟场景 */
draw( patt_trans);
/* 交换缓冲区输出成合成图像 */
argSwapBuffers();
}

```

4 实例分析

现有的样板房展示方法包括:二维平面图、实体模型等几种。二维平面图的缺点是不够生动直观,用户无法直观的感受房间的三维效果,制作实物模型不仅费时费力,而且修改起来也极为不便。

基于以上原因,笔者利用 ARToolKit 开发了一套基

于视频增强现实技术的虚拟样板房展示系统,系统能够根据自定义标识将虚拟的三维房间模型直接叠加到二维平面图上。效果如图 5 所示,其中左图为实时采集到的视频图像,右图为叠加了虚拟样板房后的效果,用户可以自由移动带有标识的二维平面图,三维虚拟样板房模型会随平面图的移动而改变位置,就像真正摆放到平面图上一样。

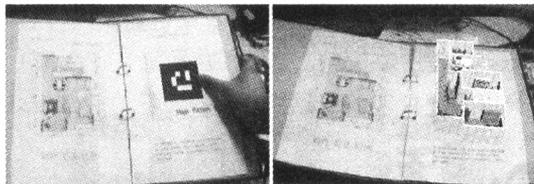


图 5 样板房展示实例

5 今后的工作

笔者利用 ARToolKit 开发了一套基于增强现实技术的样板房展示系统,今后的工作主要集中在基于增强现实技术的交互式设计与展示技术方面的研究。

参考文献

- 1 Ronald T. Azuma, A Survey of Augmented Reality [J], Teleoperators and Virtual Environments, 1997, 6 (4):355-385.
- 2 ARToolKit. <http://www.hitl.washington.edu/artool-kit/>, 2003.
- 3 Inside ARToolKit. <http://ar.in.tum.de/twiki/pub/>.