ARToolKit增强现实实践文档

增强现实技术，是一种将真实世界信息和虚拟世界信息集成的新技术，它获取现实世界的数据(可以是图像、文字，地理位置等等)，然后把计算机虚拟的数据与现实世界的数据叠加之后再呈现给用户。ARToolKit是一个开源增强现实(Augmented Reality, AR)软件库，用来快速构建增强现实应用。ARToolKit提供了多个平台下预编译的SDK，从[官网下载页面](http://www.artoolkit.org/download-artoolkit-sdk)获取安装文件。

## 相机标定

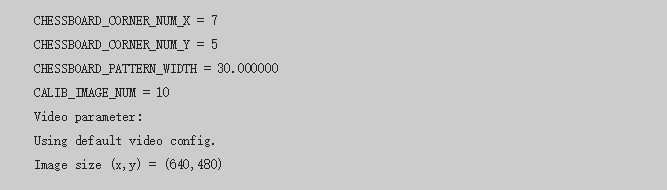
相机标定的目的是要获取摄像机成像的相关参数。常见的摄像机都基本符合“小孔成像”原理，因此需要知道其与焦距相关的参数。此外还需要估计镜头的畸变、光轴离图像中心的偏移等参数。关于相机标定的原理可以查阅计算机视觉的相关文献，这里我们只需要利用ARToolkit自带的标定工具来标定我们的摄像机。

ARToolKit软件库中默认使用的相机参数包含在相机参数文件camera\_para.dat中，每次AR应用启动时读取参数文件camera\_para.dat，此参数适用于多数应用，但是为了获得更好的精度，通常每个不同的相机需要单独标定。

ARToolKit软件库的相机标定工具calib\_camera.exe位于ARTOOLKIT5\_ROOT\bin目录，标定用的棋盘格文件ARTOOLKIT5\_ROOT\doc\patterns\Calibration\_chessboard\_(A4).pdf需提前打印到A4纸，粘贴在卡片或者板子上（使其保持平整）。

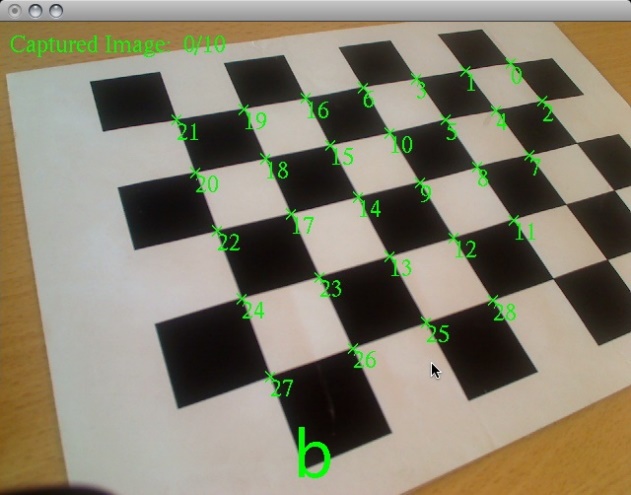
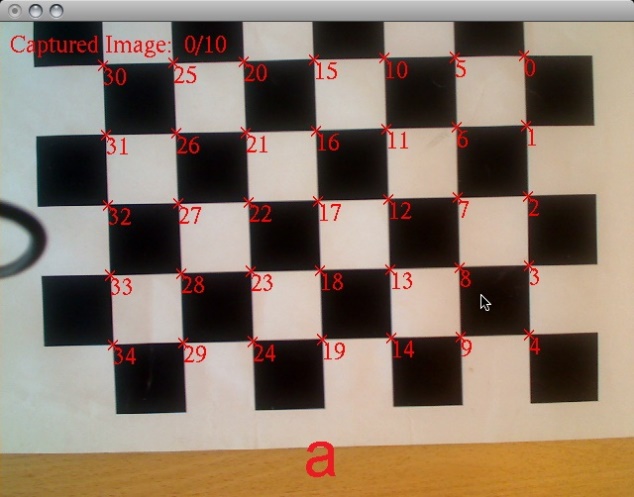
**标定过程:**

运行calib\_camera.exe，命令行终端会显示默认的参数设置

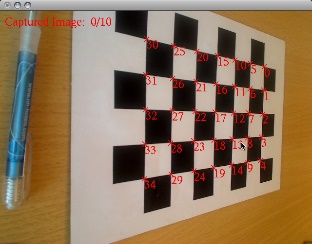
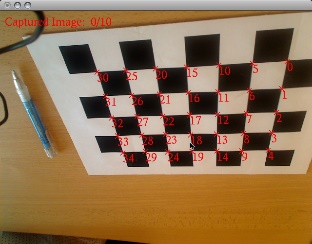
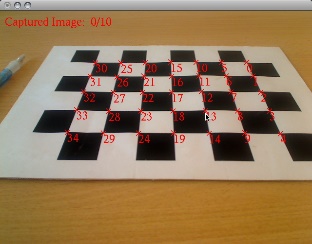
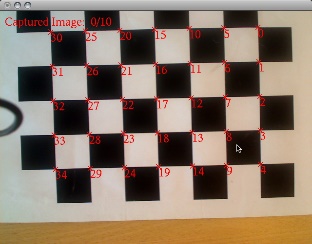


一般情况下，使用默认参数即可，若想修改参数，请输入帮助选项calib\_camera --help查看帮助提示。

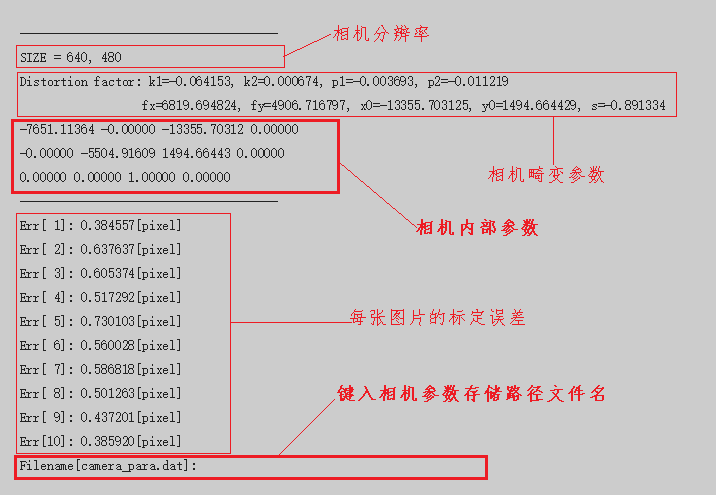
此时相机启动，用相机拍摄棋盘格，当相机拍摄到所有棋盘格内角点时，角点记号X呈红色，成功的标定画面如下图a；当相机找不到棋盘格的全部角点时，角点记号X呈绿色，如下图b



当相机捕获的图像中内角点全部呈红色时，按下键盘空格键，捕获当前帧。为了取得更好的标定效果，相机应从多个不同角度拍摄棋盘格



一旦所有的标定图像（默认10张）被获取，标定数据会输出到命令行终端并提示键入参数文件名



若标定数据良好，每张图片的标定误差应小于1像素，若误差超过2像素，应重新标定。

键入文件名，例如camera\_para.dat，回车，保存相机参数。保存的相机参数稍后将作为我们程序的输入。

## 标记的设计和训练

使用标记的目的是为了从图像估计摄像机的三维位置和姿态，即摄像机的三维跟踪。估计的位置和姿态参数将被用于虚拟物体渲染（设置虚拟视点参数），因此，精确的摄像机跟踪是增强现实应用程序的基础。

ARToolKit能够识别正方形标记（Square Marker）并在视频序列中对其进行跟踪，这是所谓的传统模板标记跟踪技术。标记往往是由用户创建或者打印出来的图案。ARToolKit软件库ARTOOLKIT5\_ROOT\doc\patterns目录下提供了一些可以直接使用的预先设计好的标记，例如Hiro\_pattern\_with\_border.pdf展示的Hiro标记，



将这些标记文件打印出来，粘贴在卡片或者板子上（使其保持平整）。

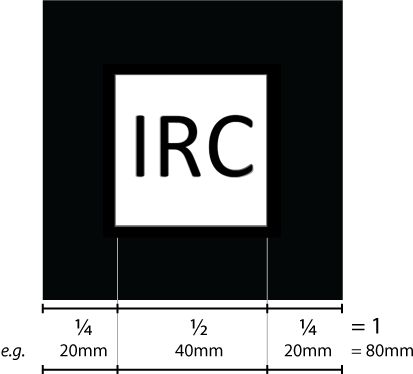
**设计新标记**

除了直接ARToolKit软件库提供的标记，我们也可以设计自己喜欢的标记，但是定制标记必须满足以下要求：

1. 必须是方形的；
2. 必须有连续的边界颜色（通常是指全黑或者全白），且其周围的背景需为对比色（通常指边界颜色的相反颜色，比如全白或者全黑）。默认情况下，边界的宽度是标记长度的25%；
3. 边界内部的标记图像不能满足旋转对称性（即不能有偶次序的旋转对称），边界内部的图像可以是白色、黑色或者其它颜色。

**设计过程**

可以通过编辑ARToolKit软件库提供的模板文件ARTOOLKIT5\_ROOT\doc\patterns\Blank\_pattern.png来创建新的标记。标记可以是任意大小的，在增强现实应用中使用标记的时，可以通过编辑配置文件指定标记相应的大小。



自定义的标记如上图所示，标记内部50%的区域被认为是标记图像。标记图像可以是彩色的、黑底白画或者白底黑画，而且可以延伸到边界区域。需要注意的是，超过标记内部50%的标记图像会被ARToolKit所忽略；因此，不要让标记图像超出边界太多，否则当相机角度倾斜较大时ARToolKit识别不出该标记。

另外一种更简单的创建标记的方式可以参考：[Julian Looser’s web-based marker generator.](http://www.roarmot.co.nz/ar/)

**训练新标记**

一旦设计好新标记，将其打印出来。接下来ARToolKit需要“训练”新标记以让其了解该标记的外形。训练过程的输出的一个图案文件，该文件包含了描述标记图像的数据。图案文件使得ARToolKit能够从场景中识别出想要跟踪的标记。在ARToolKit软件库中可以找到Hiro标记的图案文件ARTOOLKIT5\_ROOT\bin\Data\patt.hiro。

ARToolKit软件库的标记训练工具mk\_patt.exe位于ARTOOLKIT5\_ROOT\bin目录。

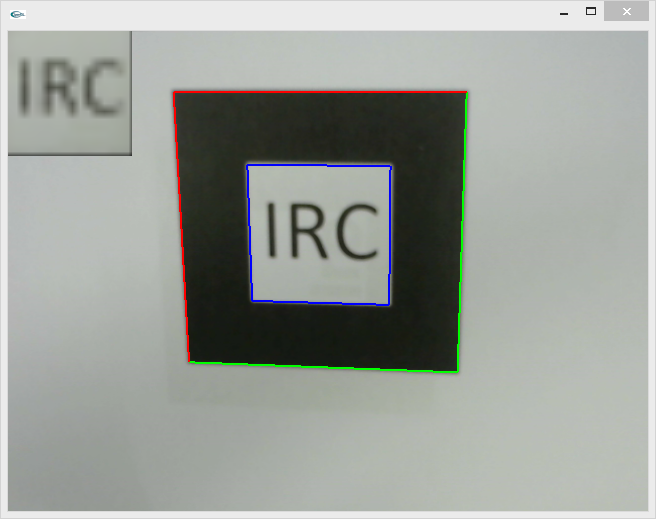
**训练过程**

运行mk\_patt.exe，会看到类似如下的提示



输入相机标定过程中保存的相机参数文件路径，回车。

此时，相机获取到视频画面



将相机对准标记，使得标记在屏幕上显示为正方形，而且尽可能大。如果ARToolKit识别出了标记，它会在标记周围画上红色或者绿色的方框线。

接下来，旋转标记使得方框红色的角位于标记的左上角，并单击左键确认。此时，标记训练完成，终端提示符/命令行提示符窗口会提示键入图案文件名



输入你的图案文件的名字（通常以“patt.name”为命名），例如patt.irc，并回车保存。如果你不想保存该文件，直接按回车来启动视频重新训练，或者单击鼠标右键退出程序。

一般在训练自定义标记（比如标记的边界大小等）时，需要指定一些参数。运行mk\_patt --help，可以查看修改默认设置的帮助提示。

**simpleARDIY - First Sample**

simpleARDIY 是一个简单的ARToolKit实践工程，包含如下C++源文件：

*DIY.h, DIY.cpp : 与ARToolkit及显示相关的核心功能*

*GLM.h, GLM.cpp : 用于读取和显示.obj模型文件的相关函数*

*simpleAR.cpp : 基于GLUT的程序界面及控制流*

将上述源文件添加到你的C++工程，然后编译运行。注意你需要首先在项目工程中完成对ARToolkit的配置，即添加包含文件、库文件及可执行文件的路径。

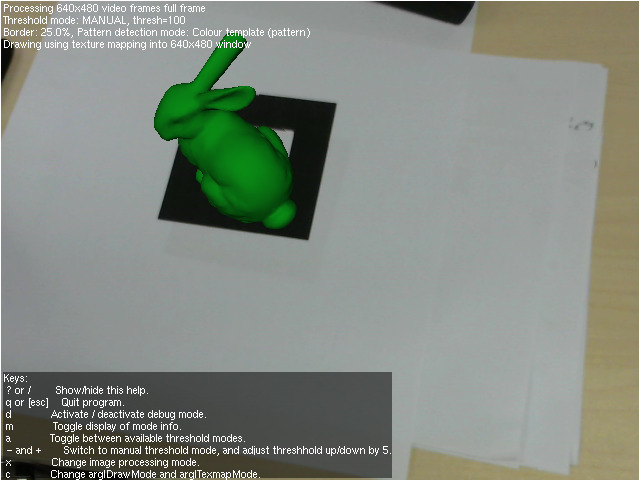
在simpleAR.cpp中定义了几个全局变量用于设置程序输入：

const char \*cparamName = "Data/camera\_para.dat"; // 相机标定参数文件

const char \*pattName = "Data/patt.irc"; // 标记模板文件

const char \*objModelFile = "Data/bunny.obj"; // .obj三维模型文件

设定好这几个参数后运行，可以得到类似这样的效果：



**DIY – Open minds and Have Fun**

以simpleARDIY工程为基础，发挥你的创意，构建你自己的增强现实应用。

See these demo videos:

[ARToolKit技术制作的坦克部队](http://v.youku.com/v_show/id_XMTYzMDkwNjQ=.html?from=s1.8-1-1.2&qq-pf-to=pcqq.c2c)

[ARToolKit增强现实游戏演示](http://v.youku.com/v_show/id_XMTAxNTE2NjQ0.html?from=s1.8-1-1.2)

**Reference**

[1] <http://www.artoolkit.org/documentation/>

[2] <http://www.hitl.washington.edu/artoolkit/documentation/>

[3] <https://github.com/artoolkit/artoolkit5>