**实验二 视觉传感器标定（实验报告）**

**【实验目的】**

1、了解开源图像处理库OpenCV的结构，掌握OpenCV的基本使用方法。

2、了解开源图像处理库OpenCV的基本模块功能，掌握常用图像处理方法。

3、掌握摄像机标定算法，学会使用OpenCV进行摄像机标定。

**【实验性质】**

验证性实验。

**【实验要求】**

1、C++集成开发环境（QT5或VS2015及以上）

2、OpenCV 4.x

3、单/双目摄像机

**【实验内容】**

1、掌握OpenCV开源图像库的基本使用方法

2、使用OpenCV进行单目摄像机标定

3、使用OpenCV进行双目摄像机标定

**【实验步骤】**

1、下载、安装、OpenCV 4.x

官网下载：https://opencv.org/opencv-4-6-0/

2、摄像机标定原理

摄像机标定的目标，是要建立三维世界坐标系与二维图像坐标系之间的对应关系。在单目视觉中，这种对应关系是一对多的，即二维图像中的一个像素点对应着三维空间中的一条直线；在双目视觉中，可以通过两幅二维图像上的对应像素点计算得到三维世界坐标系与二维图像坐标系的一一对应关系。由此便可以得到物体的三维坐标值。

摄像机安装位置参数为外部参数，摄像机镜头畸变参数为非线性模型内部参数。标定过程即是求出摄像机的内部参数和外部参数，从而得到上述坐标转换的旋转矩阵和平移向量。

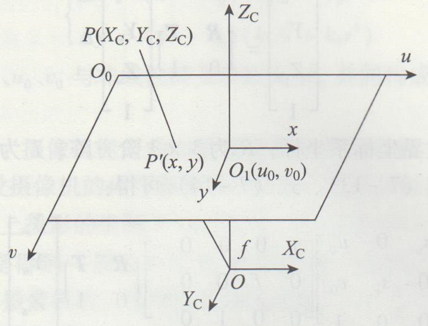


图2-1 摄像机成像模型

图中，摄像机坐标系为OXcYcZc，而计算机图像坐标系O0uv与图像平面坐标系O1xy的转换关系为：

u = sux + u0

v = svy + v0

写成矩阵形式为：

其中，su、sv分别为x、y轴方向单位长度对应的像素点数；u0、v0为镜头光学中心位置，单位为mm。

*请用矩阵形式写出计算机图像平面坐标系*O0uv*与世界坐标系的转换关系：*

|  |
| --- |
|  |

实际上，摄像机镜头成像并不是理想的透视成像，而是存在径向变形、偏心变形、薄棱镜变形等因素影响下产生不同程度的畸变。由于畸变因素作用，使空间点成像并不在线性模型描述的位置，而是在受到镜头失真影响而偏移的平面坐标。

其中，、为非线性畸变值，与像点在图像中的位置有关。一般镜头畸变同时存在径向畸变和切向畸变，切向畸变是由于透镜与摄像头传感器平面或图像平面不平行而产生的，多是由于透镜安装到镜头模组上的偏差导致，通常较小。实际应用中多只考虑径向畸变，径向畸变常用距图像中心径向距离的泰勒级数展开的前几项来表示：

、为非线性畸变参数，。

摄像机的标定方法很多，大致可分为传统标定技术和自标定技术。

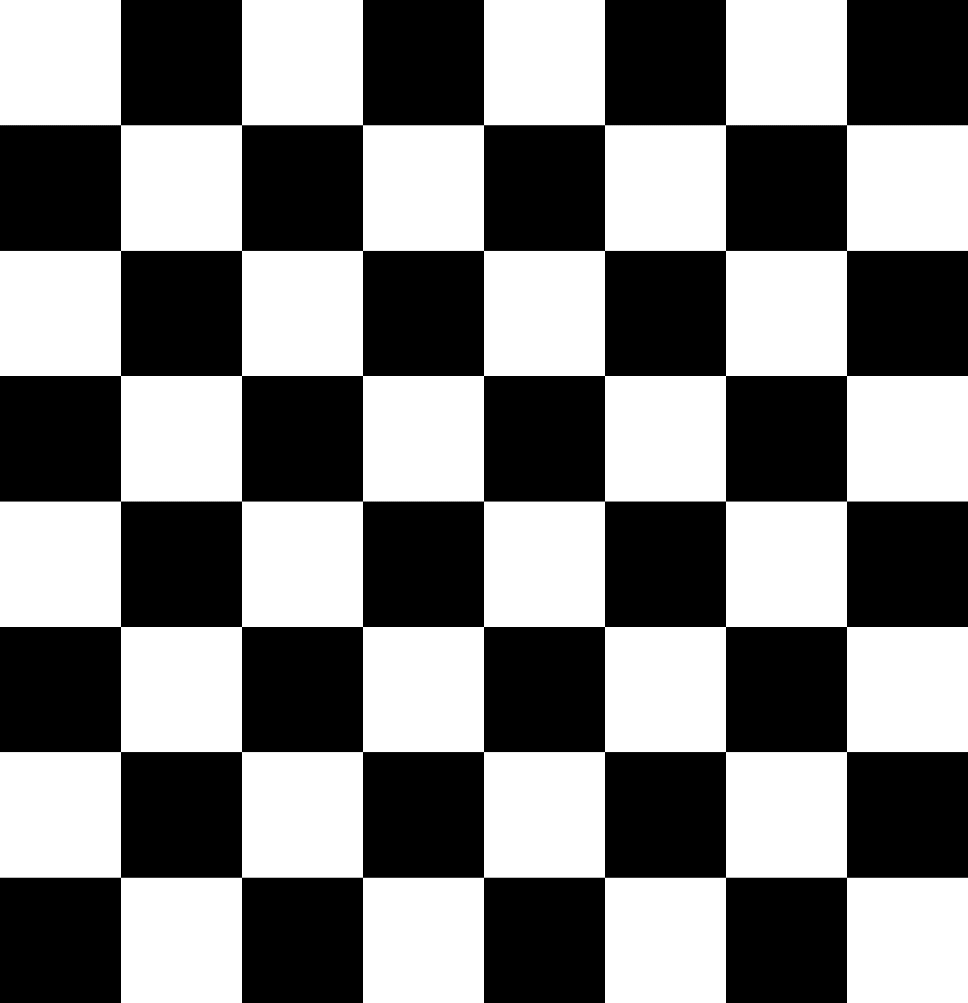
传统标定技术：

自标定技术：

OpenCV采用介于传统标定方法和自标定方法之间的一种方法，由张正友提出。这种方法不需要知道摄像机运动的具体信息，比传统标定方法更灵活，同时仍需要一个特定的标定物以及一组已知的特征基元的坐标，这一点不如自标定灵活。它通过在至少3个不同的位置获取标定物的图像，计算出摄像机所有的内外参数。

标定算法描述参考知乎：https://zhuanlan.zhihu.com/p/36371959

OpenCV源码在其sample/data目录下面一个自带的棋盘图(chessboard.png)，显示如下：



在标定的时候，算法要求提供的棋盘格的宽度与高度，还有他们的间隔距离。需要特别注意是这里的宽高是指他们的内部交叉点的个数，以上图为例，它的大小为7x7而不是8x8。间隔是指棋盘格之间的距离，可以用像素距离表示，也可以用实际毫米为单位表示。

3、标定常用函数

（1）findChessboardCorners()

bool findChessboardCorners( InputArray image, Size patternSize, OutputArray corners,

int flags = CALIB\_CB\_ADAPTIVE\_THRESH + CALIB\_CB\_NORMALIZE\_IMAGE );

函数功能：

（2）cv::drawChessboardCorners()

drawChessboardCorners( InputOutputArray image, Size patternSize,

InputArray corners, bool patternWasFound );

函数功能：

（3）find4QuadCornerSubpix()

find4QuadCornerSubpix(InputArray img, InputOutputArray corners, Size region\_size );

函数功能：

（4）cornerSubPix()

void cornerSubPix( InputArray image, InputOutputArray corners, Size winSize,

Size zeroZone, TermCriteria criteria );

函数功能：

（5）calibrateCamera()

double calibrateCamera( InputArrayOfArrays objectPoints,

InputArrayOfArrays imagePoints,

Size imageSize,

InputOutputArray cameraMatrix,

InputOutputArray distCoeffs,

OutputArrayOfArrays rvecs, OutputArrayOfArrays tvecs,

int flags = 0,

TermCriteria criteria = TermCriteria(TermCriteria::COUNT +

TermCriteria::EPS, 30, DBL\_EPSILON) );

函数功能：

（6）initUndistortRectifyMap()

void initUndistortRectifyMap(InputArray cameraMatrix,

InputArray distCoeffs,

InputArray R,

InputArray newCameraMatrix,

Size size,

int m1type,

OutputArray map1,

OutputArray map2)

函数功能：

4、单目摄像机标定

（1）打开标定配置参数文件default.xml，设置参数：

* 棋盘格的宽度和高度（两个方向的角点数量），根据实际情况设置。

<!-- Number of inner corners per a item row and column. (square, circle) -->

<BoardSize\_Width>9</BoardSize\_Width>

<BoardSize\_Height>6</BoardSize\_Height>

* 每格的宽度

单元格的宽度应设置为实际的毫米数。

<!-- The size of a square in some user defined metric system (pixel, millimeter)-->

<Square\_Size>50</Square\_Size>

* 选择输入方式

程序提供了3种输入方式。如果摄像机已连接电脑，可以使用input\_camera方式。该方式只需要设置视频输入设备号，对于笔记本而言，通常0表示笔记本内置摄像头，1表示外置摄像头。

* 编译OpenCV标定程序

（2）标定程序将实现以下功能

* 确定失真矩阵
* 确定摄像机矩阵
* 摄像机、视频和图像文件列表的输入
* 从XML/YAML文件进行配置
* 将结果保存到XML/YAML文件中
* 计算重投影误差

程序只有一个参数。其配置文件的名称。如果没有，它将尝试打开一个名为“default.xml”。

（3）运行标定程序，摄像机将拍摄25幅图片并识别角点，如下图



（4）标定结果保存在程序中指定的结果文件中

请在实验报告中说明结果文件中各项数据的意义（阅读代码，并在网上查阅相关资料）

|  |
| --- |
|  |