

单目相机标定模块 说明书

模块依赖三方库

该模块依赖 OpenCV3 图像处理库。

模块主函数介绍

caliQY::singleCameraCalibration

```
void caliQY::singleCameraCalibration(const std::string imageDir, cv::Size boardSize,  
cv::Mat& internalMatrixOut, cv::Mat& radialMatrixOut, cv::Mat& tangentialMatrixOut, int postfix);
```

这就是单目相机标定函数，函数实现以下操作：

输入标定图像所在目录和标定板棋盘内角点阵的宽长，输出计算出的内参矩阵和径向畸变向量和切向畸变向量。

例

```
int main()  
{  
    std::string imageFileDir = "data/mono";  
    imageFileDir = qy::pathJoin(currentFilePath, imageFileDir);  
  
    cv::Mat internalMatrix, radialMatrix, tangentialMatrix;  
    caliQY::singleCameraCalibration(imageFileDir, cv::Size(10, 7), internalMatrix, radialMatrix, tangentialMatrix);  
  
    return 0;  
}
```

参数

imageDir：包含标定图像的目录

boardSize：标定板棋盘内角点阵的宽长，使用的是 OpenCV cv::Size(width, height)

internalMatrixOut：计算得到的内参矩阵(输出)

radialMatrixOut：计算得到的径向畸变向量(输出)

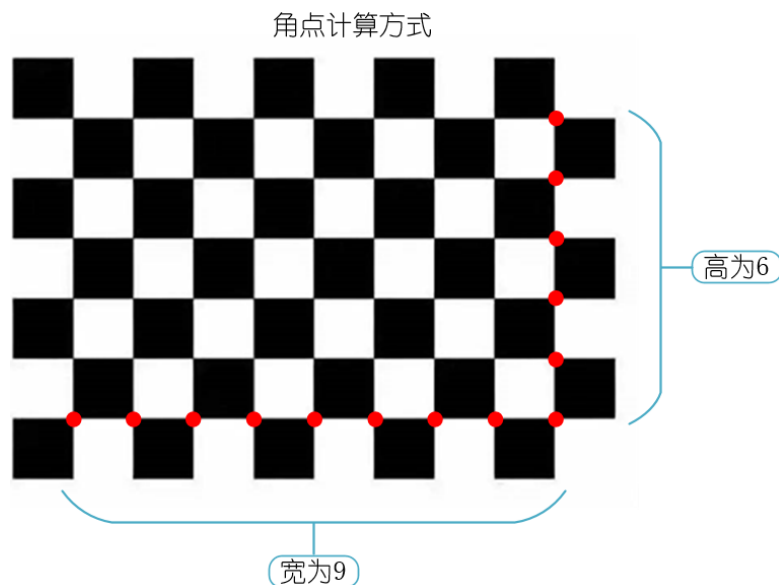
tangentialMatrixOut：计算得到的切向畸变向量(输出)

postfix: 标定图像文件的后缀名使用的是一个枚举类 `enum{JPG, PNG, JPG | PNG}`, 用于选择不同格式的图片文件（默认为 JPG）

说明

1. 标定板棋盘内角点阵怎么算？

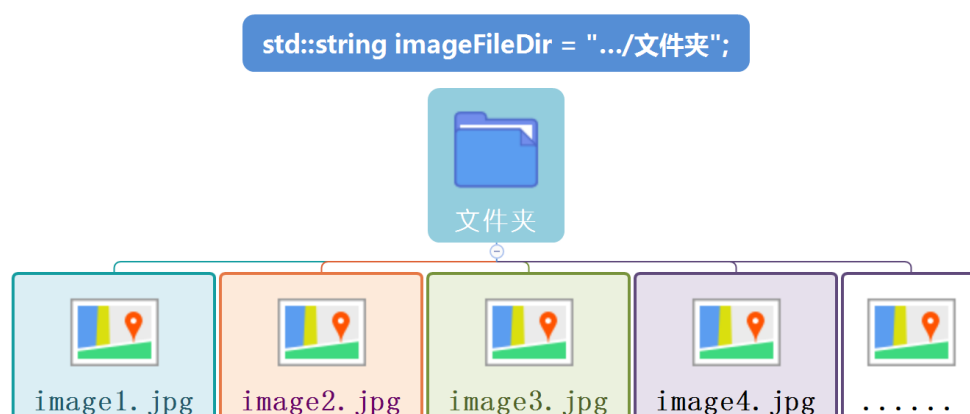
标定板棋盘内角点阵的宽和长如何计算，如下图所示。



对于这张标定板 `boardSize` 会输入 `cv::Size(9, 6)`。

2. 拍摄的标定图片怎么存放？

因为标定需要多张图片，所以将所有图片放入一个文件夹中。在函数中输入这个文件夹的路径，设定图片后缀类型，函数会自动依次读取所有图片。如下图所示。



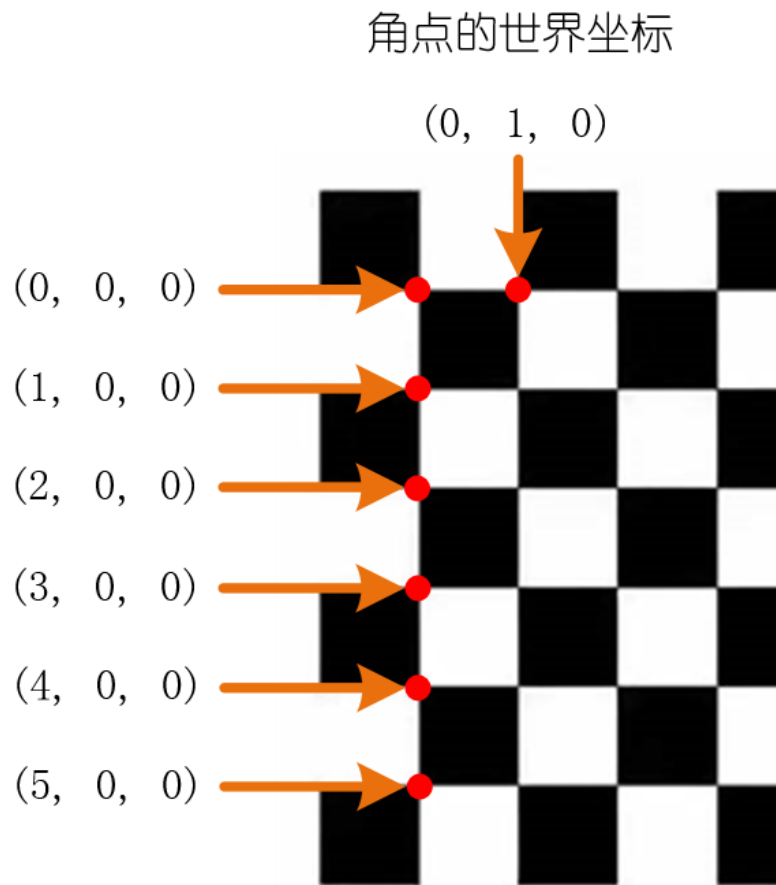
模块主函数内部的计算过程

1. 读取图片目录，获取目录下所有图片的路径，存储为路径列表。

2. 通过标定板棋盘内角点阵的宽长，计算标定板点阵的世界坐标。计算函数如下。

```
std::vector<cv::Point3f> calcObpoint(cv::Size boardSize)
```

以标定板右上角的角点为(0, 0, 0)坐标，向下行数增加，向右列数增加。坐标的第 3 个元素始终是 0。如下图所示。



3. 依次从路径列表中取出图片路径，读取图片。
4. 寻找每张图片中标定板的棋盘内角点，计算函数如下。

```
bool findChessboardCorners( InputArray image, Size patternSize, OutputArray corners,  
    int flags = CALIB_CB_ADAPTIVE_THRESH  
    + CALIB_CB_NORMALIZE_IMAGE );
```

5. 如果图片中的标定板能成功找出所有内角点，则会进一步计算将角点精确到亚像素级，以便之后能计算出更加精确的标定结果。计算函数如下。

```
void cornerSubPix( InputArray image, InputOutputArray corners,  
    Size winSize, Size zeroZone,  
    TermCriteria criteria );
```

6. 对于所有成功检测到棋盘内角点的标定板图片，将图片中标定板的角点像素坐标和对应的世界坐标记录用于之后的计算。(因当前标定只使用了一张标定板，且标定板中途不会更换，所以世界坐标始终是第 2 步设定的坐标)
7. 将角点像素坐标集和对应的世界坐标集输入到标定函数中计算，从而得到标定结果。计算函数如下。

```
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) );
```