[IplImage](http://baike.baidu.com/subview/3083269/3083269.htm" \t "http://baike.baidu.com/_blank)\* cvQueryFrame( CvCapture\* capture ); cvQueryFrame从摄像头或者文件中抓取一帧，然后解压并返回这一帧。

void assert( int expression );现计算表达式 expression ，如果其值为假（即为0），那么它先向stderr打印一条出错信息，然后通过调用 abort 来终止程序运行.

CvMat\* cvCreateMat(int rows, int cols, int type);例如: CV\_8UC1 表示8位无符号单通道矩阵, CV\_32SC2表示32位有符号双通道矩阵. 例程: CvMat\* M = cvCreateMat(4,4,CV\_32FC1);

FindChessboardCorners 寻找棋盘图的内角点位置 int cvFindChessboardCorners( const void\* image, CvSize pattern\_size, CvPoint2D32f\* corners, int\* corner\_count=NULL, int flags=CV\_CALIB\_CB\_ADAPTIVE\_THRESH ); image 输入的棋盘图，必须是8位的灰度或者彩色图像。 pattern\_size 棋盘图中每行和每列角点的个数。 corners 检测到的角点 **corner\_count 输出，角点的个数。如果不是NULL，函数将检测到的角点的个数存储于此变量。** flags 各种操作标志，可以是0或者下面值的组合： CV\_CALIB\_CB\_ADAPTIVE\_THRESH - 使用自适应阈值（通过平均图像亮度计算得到）将图像转换为黑白图，而不是一个固定的阈值。 CV\_CALIB\_CB\_NORMALIZE\_IMAGE - 在利用固定阈值或者自适应的阈值进行二值化之前，先使用cvNormalizeHist来均衡化图像亮度。 CV\_CALIB\_CB\_FILTER\_QUADS - 使用其他的准则（如轮廓面积，周长，方形形状）来去除在轮廓检测阶段检测到的错误方块。 函数cvFindChessboardCorners试图确定输入图像是否是棋盘模式，并确定角点的位置。如果所有角点都被检测到且它们都被以一定顺序排布（一行一行地，每行从左到右），函数返回非零值，否则在函数不能发现所有角点或者记录它们地情况下，函数返回0。例如一个正常地棋盘图右8x8个方块和7x7个内角点，内角点是黑色方块相互联通地位置。这个函数检测到地坐标只是一个大约地值，如果要精确地确定它们的位置，可以使用函数cvFindCornerSubPix。

函数cvFindCornerSubPix()用于发现[亚像素](http://baike.baidu.com/subview/2615333/2615333.htm" \t "http://baike.baidu.com/_blank)精度的[角点](http://baike.baidu.com/subview/4051484/4051484.htm" \t "http://baike.baidu.com/_blank)位置[1]  。

void cvFindCornerSubPix(

const CvArr\* image,

CvPoint2D32f\* corners,

int count,

CvSize win,

CvSize zero\_zone,

CvTermCriteria criteria

);

image

输入的图像，必须是8位的[灰度](http://baike.baidu.com/subview/974296/974296.htm" \t "http://baike.baidu.com/_blank)或者彩色图像。

corners

输入[角点](http://baike.baidu.com/subview/4051484/4051484.htm" \t "http://baike.baidu.com/_blank)的初始坐标，也存储精确的输出坐标。

count

角点数目

win

搜索窗口的一半尺寸。如果win=（5,5）那么使用（5\*2+1）×（5\*2+1）=11×11大小的搜索窗口

zero\_zone

死区的一半尺寸，死区为不对搜索区的中央位置做求和运算的区域。它是用来避免[自相关矩阵](http://baike.baidu.com/subview/5744115/5795525.htm" \t "http://baike.baidu.com/_blank)出现的某些可能的奇异性。当值为（-1，-1）表示没有[死区](http://baike.baidu.com/subview/421246/421246.htm" \t "http://baike.baidu.com/_blank)。

[criteria](http://baike.baidu.com/subview/1813717/1813717.htm" \t "http://baike.baidu.com/_blank)

求角点的[迭代](http://baike.baidu.com/subview/461623/461623.htm" \t "http://baike.baidu.com/_blank)过程的终止条件。即角点位置的确定，要么迭代数大于某个设定值，或者是精确懂达到某个设定值。criteria可以是最大迭代数目，或者是设定的[精确度](http://baike.baidu.com/subview/793478/793478.htm" \t "http://baike.baidu.com/_blank)，也可以是它们的组合。

cvDrawChessboardCorners(

IntPtr image,

Size patternSize,

IntPtr corners,

int count,

int patternWasFound

) 用于标定摄像机时绘制被成功标定的角点

image——输入的目标图像，必须是8位彩色图像

patternSize——内心的角落每棋盘的行和列的数目角，也就是标定板角点的行数和列数

corners——检测到的角阵列，也就是用cvFindCornerSubPix函数检测到的角点的坐标（cvFindCornerSubPix的第二个参数）

count——角数，每张标定图所有角点的数目

patternWasFound——指出是否已找到所有的角点（该值为0表示不能找到所有的角点，不为0则表示能够找出所有的角点），该参数值由cvFindChessboardCorners函数的返回值给出