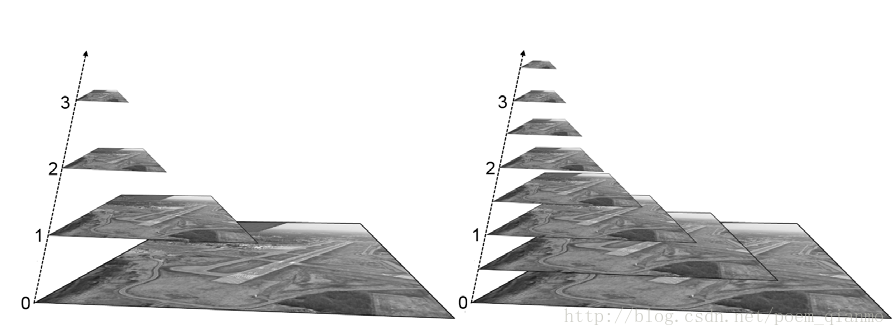
**ORB-SLAM2 庖丁解牛之A-1-(1)**

上一回说到ORB Extraction，而首先就是要对FAST角点进行提取，但是在代码中，ORBSLAM先把图像金字塔做了，为什么？因为需要对每层的图像进行FAST调用。

* 先理论基础：

那么，我们先了解一下，图像金字塔是啥东西？

图像金字塔是图像中多尺度表达的一种，是一种以多分辨率来解释图像的有效但概念简单的结构。一幅图像的金字塔是一系列以金字塔形状排列的分辨率逐步降低，且来源于同一张原始图的图像集合。其通过梯次向下采样获得，直到达到某个终止条件才停止采样。金字塔的底部是待处理图像的高分辨率表示，而顶部是低分辨率的近似。我们将一层一层的图像比喻成金字塔，层级越高，则图像越小，分辨率越低。就像这样：

 有什么作用呢？

图像金字塔是图像中多尺度表达的一种，最初用于机器视觉和图像压缩，最主要用于图像的分割、融合。

怎么实现？

而理论上，一般有2种经典金字塔：高斯金字塔和拉普拉斯金字塔。因为ORBSLAM用到高斯图像金字塔，所以只讲高斯金字塔。

高斯金字塔 ( Gaussian pyramid)是由底部的最大分辨率图像逐次向下采样得到的一系列图像。最下面的图像分辨率最高，越往上图像分辨率越低。

注意：降采样其实是由金字塔底部向上采样，分辨率降低，它和我们理解的金字塔概念相反。

* 后工程实践：

在OpenCV中，有关图像金字塔的操作有pyrup()和pyrdown()，分别用来完成上采样和下采样，其实还有更直接的函数，即resize()，而ORBSLAM中使用的是resize()。怎么用的呢？来看看之前撒的面包屑：

|  |  |  |
| --- | --- | --- |
| 函数 | 函数（步骤） | 作用 |
| ORBextractor::operator() | getMat | 获取图像 |
| ComputePyramid | 计算图像金字塔 |
| ComputeKeyPointsOctTree | 利用四叉树均匀化特征点 |
| GaussianBlur | 高斯模糊 |
| computeDescriptors | 计算高斯模糊后图像的描述子 |

在ComputePyramid函数实现图像金字塔构建。其中resize实现图像缩放。

resize(mvImagePyramid[level-1],

//输入图像

mvImagePyramid[level],

//输出图像

sz,

//输出图像的尺寸

0,

//水平方向上的缩放系数，留0表示自动计算

0,

//垂直方向上的缩放系数，留0表示自动计算

cv::INTER\_LINEAR

//图像缩放的差值算法类型，这里的是线性插值算法

);

贴样例代码：

要讲高斯金字塔吗？好像后面讲会比较好，因为现在还没到那一步。