<http://www.cnblogs.com/zsb517/archive/2012/07/04/2575847.html>

codebook采用量化技术从时间序列中获得背景模型，能够检测像素剧烈变化、或者有移动物体或者更为复杂的背景模型。codebook为每个像素建立一个codebook，每个codebook含有一个或者多个codeword，codeword 记录背景学习的阈值、对应像素的更新时间以及访问频率等，通过这些信息，可以得知每个像素的变化情况，从而获得视频中的背景模型。

1.opencv实现简单Codebook

CodeBook算法为当前图像的每一个像素建立一个CodeBook(CB)结构,每个CodeBook结构又由多个CodeWord(CW)组成。CB和CW的形式如下：

CB={CW1,CW2,…CWn,t}

CW={lHigh,lLow,max,min,t\_last,stale}

其中n为一个CB中所包含的CW的数目，当n太小时，退化为简单背景，当n较大时可以对复杂背景进行建模;t为CB更新的次数。CW是一个6元组，其中IHigh和ILow作为更新时的学习上下界，max和min记录当前像素的最大值和最小值。上次更新的时间t\_last和陈旧时间stale(记录该CW多久未被访问)用来删除很少使用的CodeWord。

假设当前训练图像I中某一像素为I(x,y)，该像素的CB的更新算法如下，另外记背景阈值的增长判定阈值为Bounds：

(1) CB的访问次数加1；

(2) 遍历CB中的每个CW，如果存在一个CW中的IHigh，ILow满足ILow≤I(x,y)≤IHigh，则转(4)；

(3) 创建一个新的码字CWnew加入到CB中, CWnew的max与min都赋值为I(x,y)，IHigh <- I(x,y) + Bounds，ILow <- I(x,y) – Bounds，并且转(6)；

(4) 更新该码字的t\_last，若当前像素值I(x,y)大于该码字的max，则max <- I(x,y)，若I(x,y)小于该码字的min，则min <- I(x,y)；

(5) 更新该码字的学习上下界，以增加背景模型对于复杂背景的适应能力，具体做法是：若IHigh < I(x,y) + Bounds，则IHigh 增长1，若ILow > I(x,y) – Bounds，则ILow减少1；

(6) 更新CB中每个CW的stale。

使用已建立好的CB进行运动目标检测的方法很简单，记判断前景的范围上下界为minMod和maxMod，对于当前待检测图像上的某一像素 I(x,y)，遍历它对应像素背景模型CB中的每一个码字CW，若存在一个CW，使得I(x,y) < max + maxMod并且I(x,y) > min – minMod，则I(x,y)被判断为背景，否则被判断为前景。

在实际使用CodeBook进行运动检测时，除了要隔一定的时间对CB进行更新的同时，需要对CB进行一个时间滤波，目的是去除很少被访问到的CW，其方法是访问每个CW的stale，若stale大于一个阈值（通常设置为总更新次数的一半），移除该CW。

利用opencv实现：代码

http://www.cnblogs.com/Images/OutliningIndicators/ContractedBlock.gif

codebook实现例子

该算法测试的时候，对背景的变换有一定的适应性，但是获取的前景目标空洞比较多，需要后面的区域处理上做一些功夫。opencv上面的例子，只是使用图像的亮度作为参数来更新背景模型，对阴影、环境光线的变化等等 处理的并不好。