*本文档主要简述运动估计，相对运动的基本方式：*

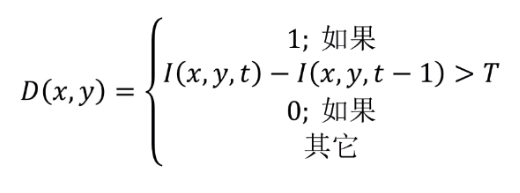
* *背景提取(减除) —— 相机静止，目标运动*
* *光流估计(全局运动) —— 相机运动，目标静止*
* *光流估计 —— 相机和目标均运动*

# 背景提取（减除）

背景提取（减除）主要处理相机静止，目标运动的场景。

## 帧差法运动目标检测

### 帧差法原理



对应像素相减，若差值大于一个阈值，则置为1；反之置为0。

 D(x, y): 帧差

 I(x,y,t): 当前帧(t时刻)图像

 I(x,y,t): 上一帧(t-1时刻)图像

 T: 像素灰度差阈值

### 帧差法问题

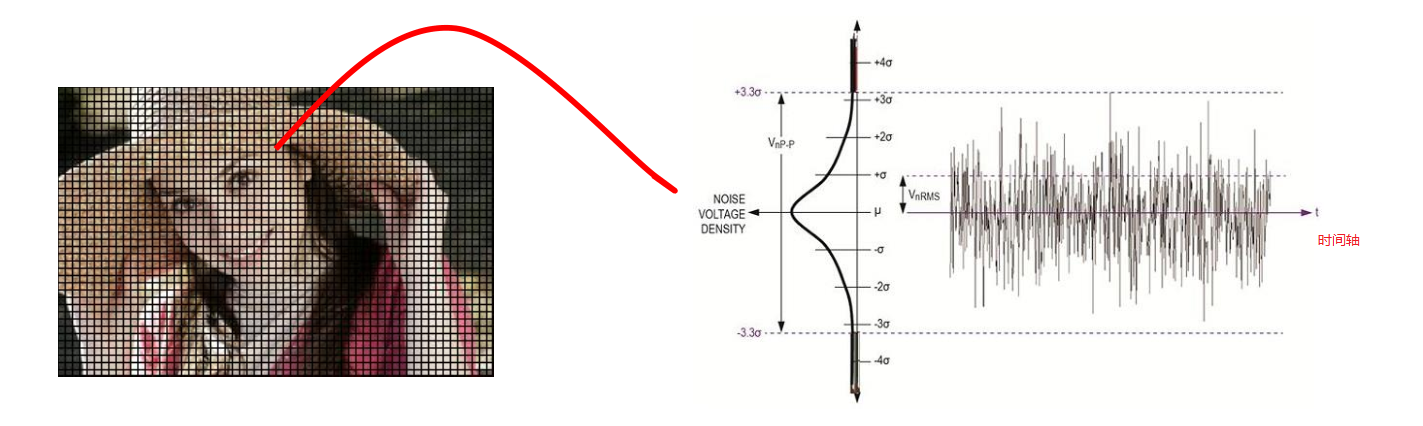
相邻两帧图像存在目标存在重合，会存在空洞现象；

图像本身存在噪声，相减可能会放大噪声

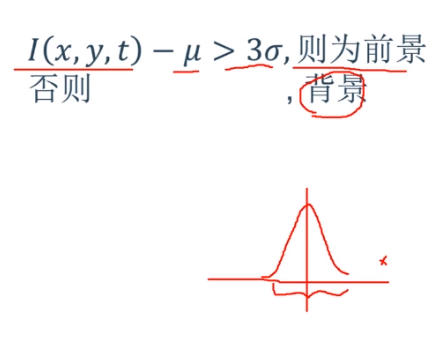
## 高斯背景建模

### 高斯背景建模的基本原理

由于环境中光照影响，图像中像素值不会一直保持不变，其像素灰度值随时间变化符合高斯分布，如下图：

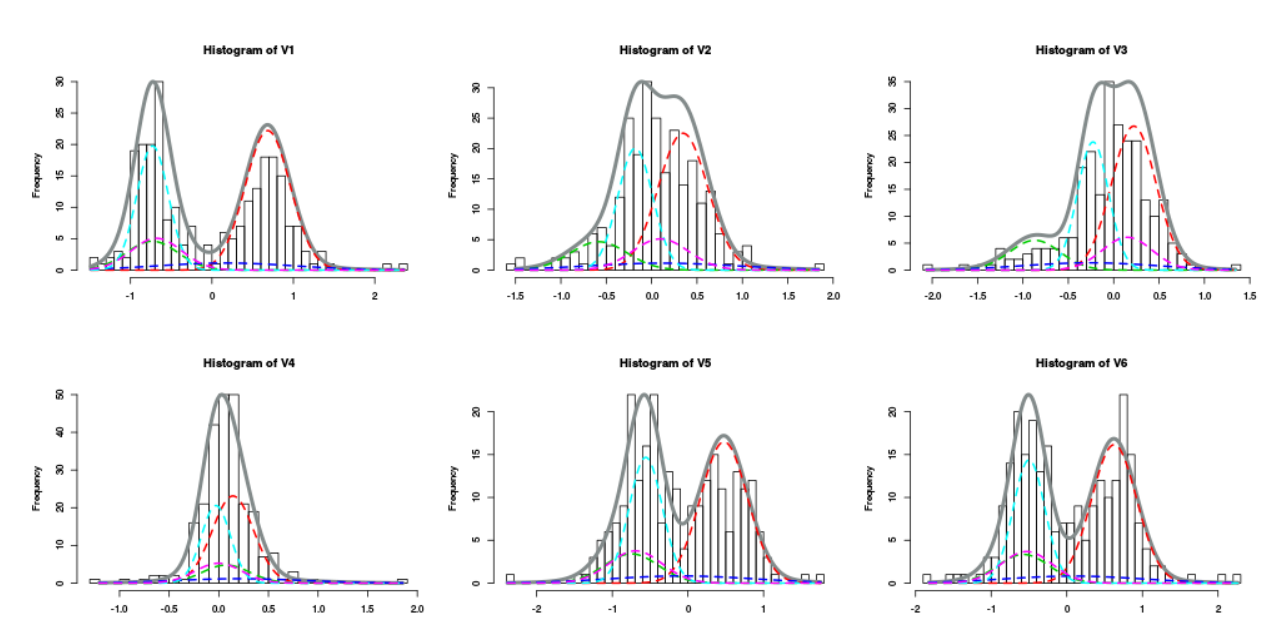


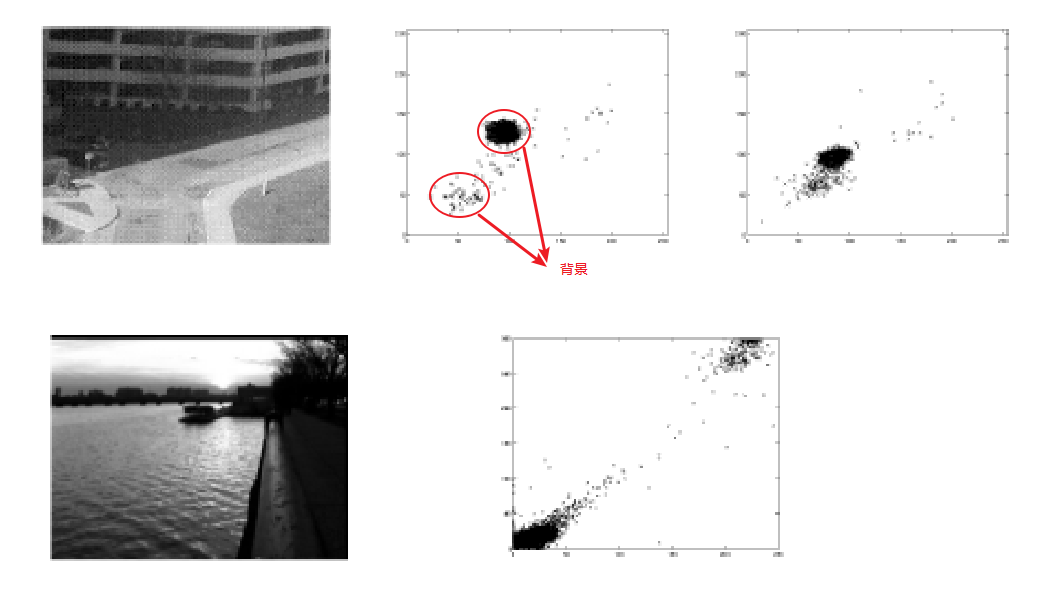
根据经验得，一般情况下像素波动范围在3σ内（如上图），所以判断一幅图像：

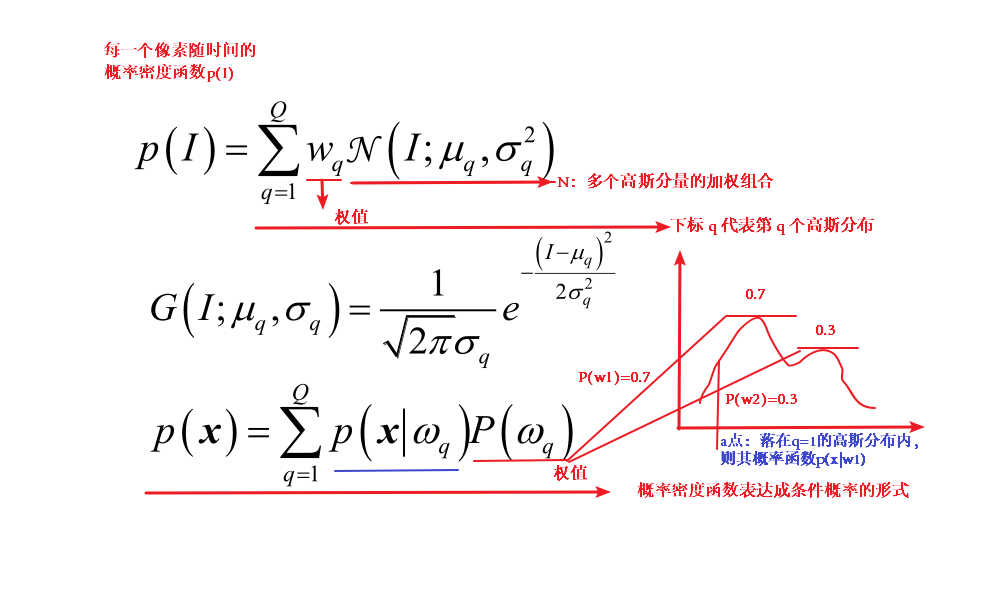


**混合高斯模型：**

有些分布不能用一个高斯分布判断，但可以看做多个高斯模型的混合。如下图:





像素灰度(随时间)的概率密

### 混合高斯模型建模步骤（视频数据）：

1. 模型初始化 将采到的第一帧图像的每个象素的灰度值作为均值，再赋以较大的斱差。初值 Q =1, w=1.0。（Q是高斯模型的分量个数，此时因只有一个图像，所以Q=1；w为权值）

假设图像中点a(x,y)的灰度值为50，那么设第一个高斯图像的均值u为50，w=1，方差σ可随便设如σ=10，这样就得到了第一个高斯分布。（计算所有像素点的高斯分布函数？）

1. 模型学习 将当前帧的对应点象素的灰度值与已有的 Q 个高斯模型作比较，若满足(满足此函数表示其值落在第Q个高斯函数内部)，则按上页方式调整第q个高斯模型的参数和权重；否则转入(3)：

此时假设第二帧图像其点a(x,y)的灰度值为53，若53落在了第一步中的高斯分布函数内，那么进行调整：

均值u= （50+53）/2=51.5 ；方差σ= (51.5-50)² + (51.5 – 53) ² = 4.5

权重的调整可参看混合高斯模型迭代计算原理

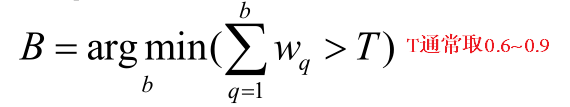
1. 增加/替换高斯分量 若不满足条件，且 q < Q ，则增加一个新分量；若q=Q (通常Q=3~5)，则替换

此时假设第三帧图像其点a(x,y)的灰度值为180，没有落在第Q个高斯模型内，那么进行增加：

此时q=1， q<Q，则使用此时点按步骤1再建立一个高斯分量；此时就有了2个高斯分量，q=2的高斯分量其均值就是180，σ=10。

若经过多次处理，我们已经有了5个高斯分量（若Q=5），新的图像不在任何一个高斯分量内，则进行替换：

则将权重最小的高斯模型替换为以时计算出的高斯模型

1. 判断背景

假设经过了1000帧图像，背景模型以基本建立：其五个高斯分量分量权值分别为：0.3、0.4、0.06、0.18、0.06

第一步先将5个高斯分量按权值从大到小进行排序：0.4、0.3、0.18、0.06、0.06

计算B，若T取0.6，则从最大到最小的顺序累加权值使得累加值大于T，此时计算为：0.4+0.3 > 0.6;则b=2。

那么我们认为：若其像素的灰度值属于0.4或0.3的，我们认为它为背景点，否则为前景。

（0.18、0.06、0.06这三个权值的高斯分量我们认为是噪声）

1. 判断前景

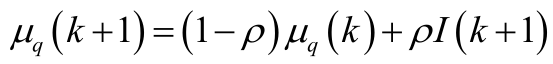
### 混合高斯模型迭代计算原理

**权值w：**



* w(k+1)为当前帧权值
* w(k)为上一帧权值
* M(k+1)为二值化函数：若当前像素属于我们当前处理的这个分布，则M为1，否则为0
* α对应的是学习速率，取(0~1)之间的数；由上式可以看出，α越大则越小，那么其上一帧图像的的权值对这一帧图像权值影响也就越小，同时本帧数据对本帧权值影响也就越大。
* α的取值一般在0.005~0.05之间的数
* 类别通常取3~5之间的数（q）

**均值q：**

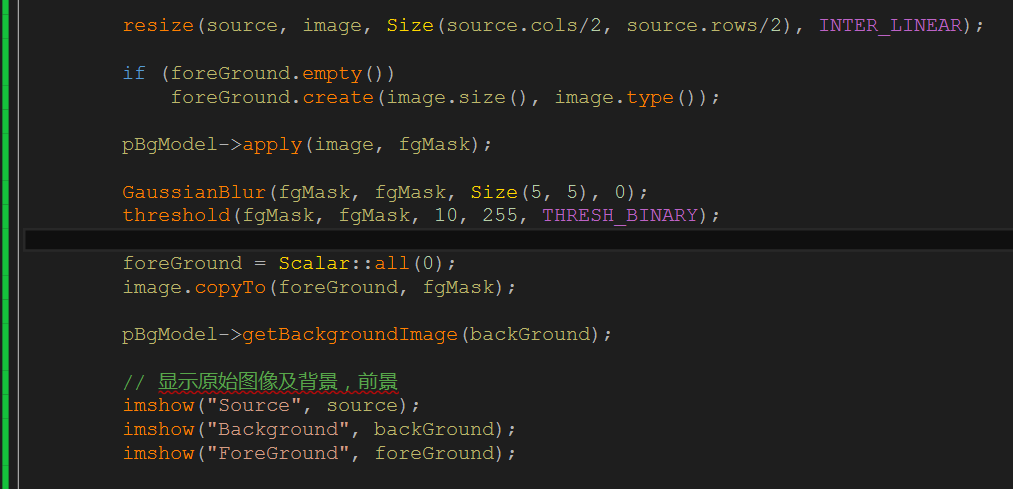


ρ也是学习速率，取(0~1)之间的数,类似α

**方差σ：**



## 背景建模OpenCV相关函数：



# 光流估计

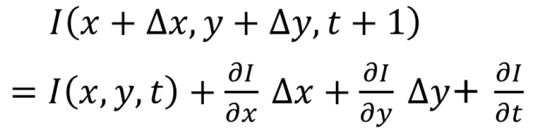
***光流估计解决相机运动的场景***

## 光流估计的基本方程

光流估计解决当相机运动时，两帧图像中像素移动位置。

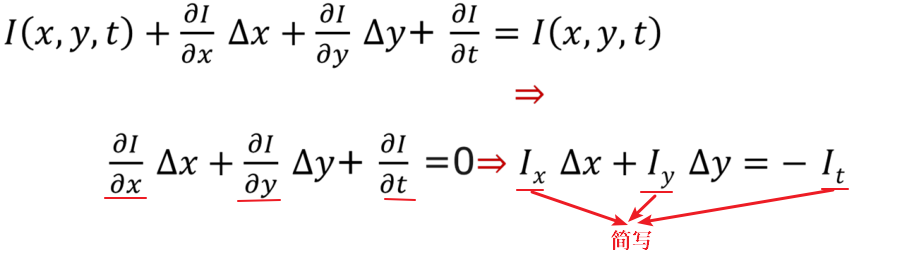
### 光流估计的基本模型

在每一个像素(x,y,t)处，有;意义为：t时刻像素(x,y)的点在t+1时刻移动到了(x+△x，y+△y)处。对进行泰勒展开：



* I(x,y,t)表示：t时刻在（x，y）像素点的亮度值
* 若要适用泰勒展开，那么△x和△y必须充分的小，其次、、这三个偏导项是存在的。这句话的在实际场景中表示相邻两帧图像变化不大，实际情况也是如此。

**因此：**



* 、、这三个偏导项简写成Ix，Iy，It
* 当给定两帧图像是Ix，Iy，It这三个偏导数是可求的，可以用差分来计算
* △x和△y就是像素的位移值，也就是所求的光流

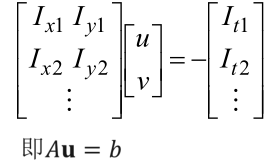
光流就是上式中的△x和△y，下节我们讨论L-K光流估计方法

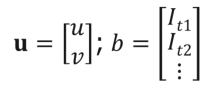
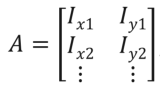
## L-K光流估计方法

### 如何求光流（△x和△y）

用**u、v表示△x和△y**；由于，一个小方格里的所有像素位移相同，那么可以组成列成多个表达式，如：；

可用矩阵表示：如下



* 其中
* 这里的(Ix1，Iy1)、(Ix2，Iy2)…，若3\*3邻域则为9组，每一组表示一个像素点的计算式

最优化问题(超定方程求解)



最小二乘解：



* 区域像素只有2个时，就是2元1次方程组求解！多个像素，比如3 ∗ 3时，则是求上述最小二乘解

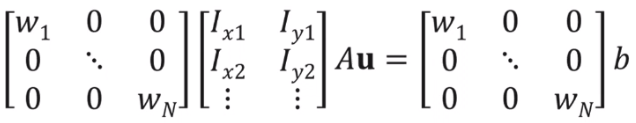
总结一下：L-K光流估计方法的思路就是在图像的小的邻域内其速度是一致的

其公式如下：



* wi表示权重，比如3\*3邻域内，要求中心点的速度，那么越靠近中心其**△x和△y**比重越大w。
* w可取成高斯函数取值

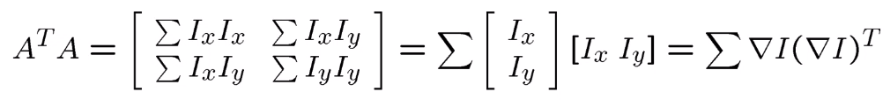
加上权重w，式可变为：

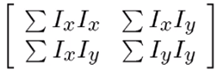


同样适用最小二乘的方法求：，此式有矩阵的逆，但不是所有矩阵都有逆，需要判断矩阵是否可逆。

### 矩阵求逆是否能实现（可信度判断）

**根据x方向y方向灰度梯度值判断**



* 如果Ix和Iy有一项为0的话，那么是不可逆的。那么Ix和Iy有一项为0，说明图像在x方向或y方向灰度值变化较平坦。

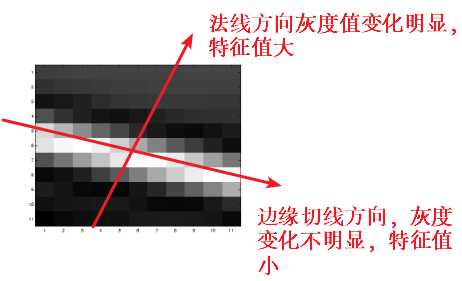
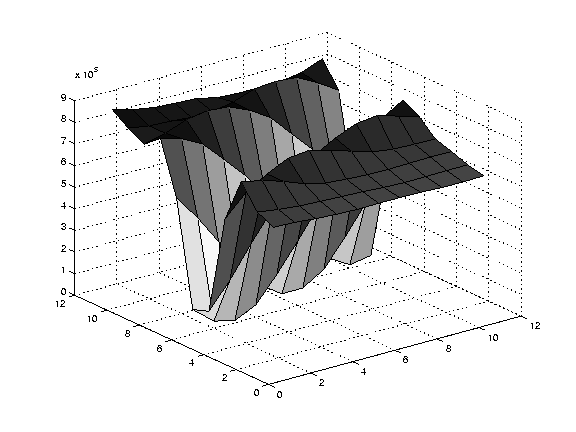
**通过特征值判断**

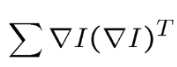
若存在一个特征值，其值接近于0，则不可逆。

### L-K方法应用场景

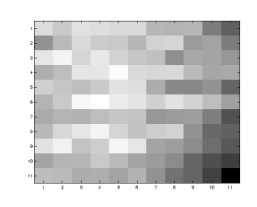
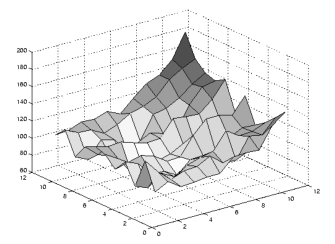
**图像边缘求光流**

边缘(对比Harris算子)，如下图：

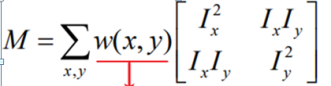
 

：沿某一方向剧烈变化，则沿某方向灰度值小，矩阵不可逆，也就无法求出光流。

**低纹理区域(对比Harris算子)**

** **

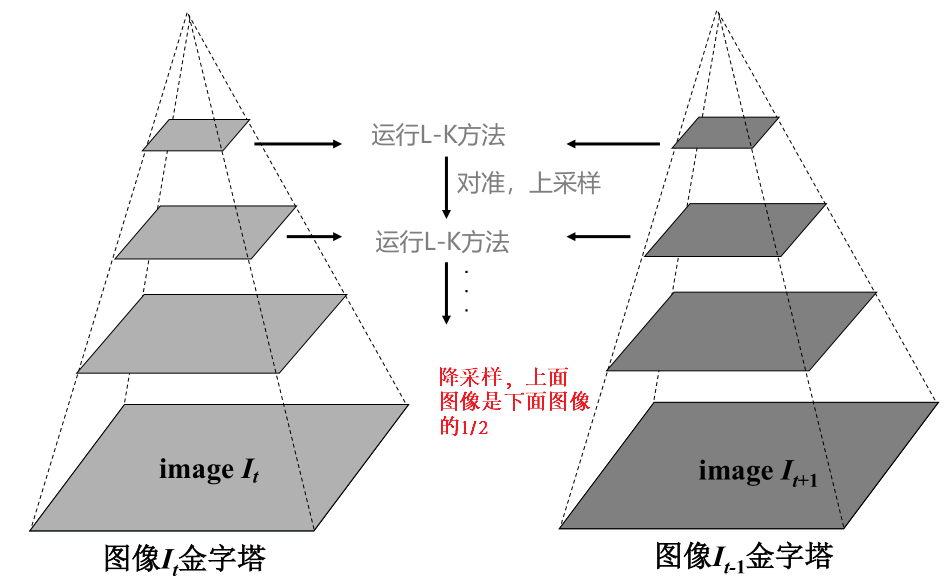
底纹理区各个方向变化均不明显，所以其特征值都较小，也无法求光流。

可看角点检测Harris算子公式()，与光流计算公式相似，分析可知只有在角点处在各方向灰度变化较大，L-K方法在角点处可求光流；而角点在一幅图像中占图像中的0.5%~5%，比较少，所以L-K方法有叫稀疏光流算法。

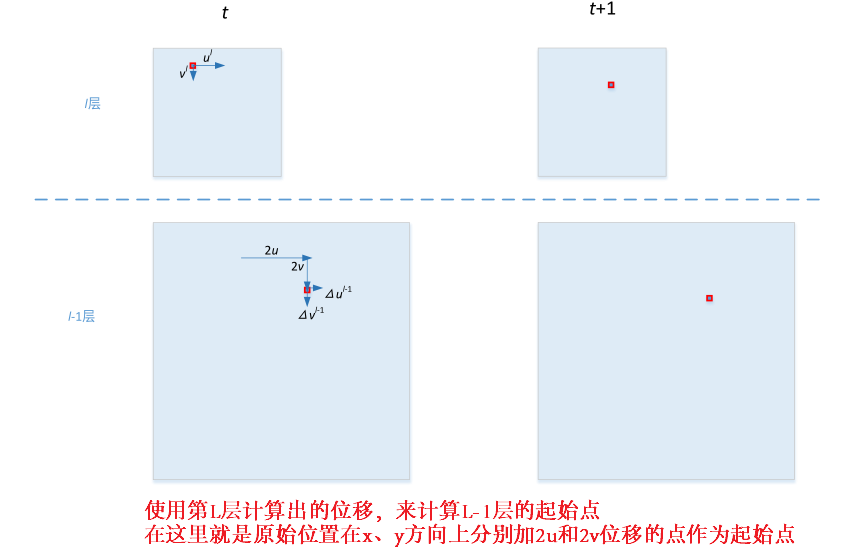
## 金字塔L-K方法

原始L-K光流方法是计算邻域内像素，若像素位移较大，则不适用。我们可以使用金字塔L-K方法来处理大位移场景；金字塔L-K方法对噪声不敏感。

具体处理方式就是将图像进行1/2降采样，构造出金字塔模型，那么原先10个像素的位移，在下一层降采样图像上只有5个像素；不断进行降采样可以缩小位移，在最上层降采样图像上可能只有1个像素位移，我们通过降采样图像上计算出的光流在对应到底层图像上，最终获取大位移场景下的光流。



**金字塔光流位移传播示意图**



## 总结

1. 光流估计基于恒定亮度假设模型

2. L-K光流估计方法利用了邻域内的运动不变性

3. 金字塔L-K方法可有效提高光流计算对大位移的鲁棒性