# LK和DIS算法对比分析报告

## 算法流程

### LK(Lucas-Kanade)

假设I和J是连续的两帧灰度图。分别表示I和J中点处的灰度值。左上角点坐标为，右下角点坐标为。现在考虑I中的一点，现在目标是在J中找到一点，使得I(u)和J(v)相同（相似）。其中称之为I中点X处的光流。以X为中心选取大小为的窗口，假设窗口内的像素点具有相同的光流，则d为下面最小化优化问题的解：

#### 图像金字塔

定义图像的大小为，表示第0层图像，则第0层图像的长和宽为，。

图像金字塔通过递归的方式建立：由计算得到，由计算得到……假设是一个图像金字塔，则有

在计算前先将()周围填充1个像素。

易知：

式子(3),(4),(5)用来计算图像I和J的图像金字塔，得到：和。式(3)在使用前先对图像低通滤波。滤波核为。

#### 金字塔特征跟踪

定义u为原图I中的点，为金字塔中第层图像中与u对应的点。则：

金字塔光流计算算法如下，首先计算金字塔最上层的光流，,然后将计算结果传播到第层作为该层像素u位移的初始值，在这个初始值的基础上计算层的光流，然后将结果传到层，以此递归计算到第0层。

现在假设是第L层图像的初始光流位移量，是第L层通过最小化损失函数得到的光流残差。

则有：

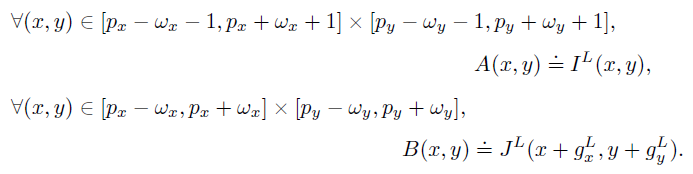
其中：

最终光流d为：

这样在每一层都只要计算一个小的光流，结果可以构成一个大光流d。

#### 迭代LK光流计算过程

在金字塔每一层的目标是求解使(6)最小。定义



令

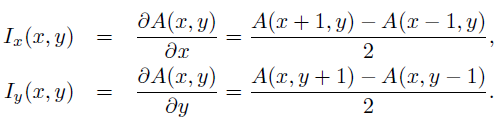
则：

对求导令导数等于0可解得：

其中：

又有：

C:\Users\hudawei\AppData\Roaming\Tencent\Users\526451080\QQ\WinTemp\RichOle\LUI4VDGO$2UNP`0@`(M02KN.png



#### 算法流程

## 算法运行时间

## 结果