

第10章 光流计算



计算机科学与技术学院

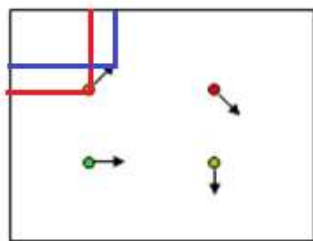
本次课程内容

1. 什么是光流
2. 光流估计的用途
3. 运动场与光流关系
4. 光流的估计方法
5. 密集光流与稀疏光流

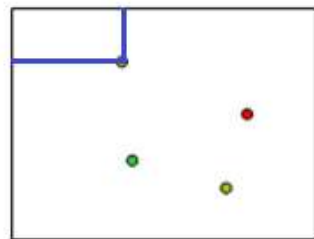


1. 什么是光流 (optical flow)

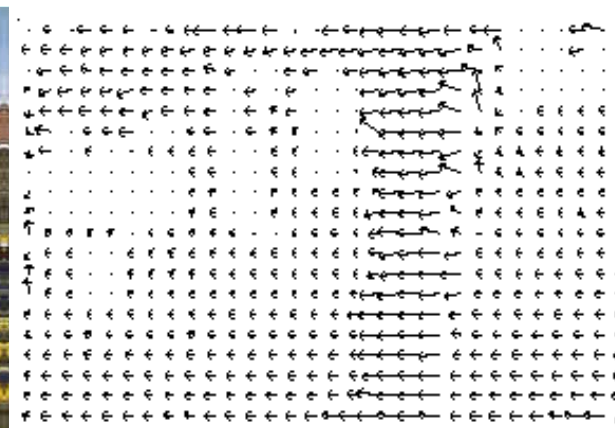
◆ 光流: 是空间运动物体在成像平面上像素运动的瞬时速度



前一帧



后一帧



from S. Seitz, R. Szeliski, M. Pollefeys

什么是光流 (optical flow)

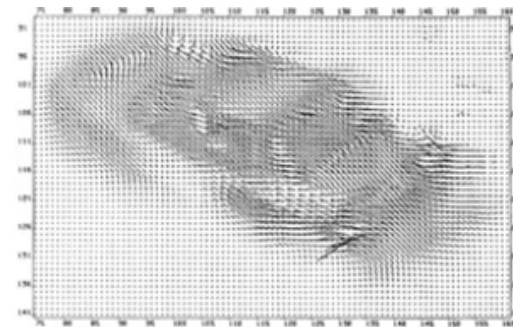


计算光流的目的

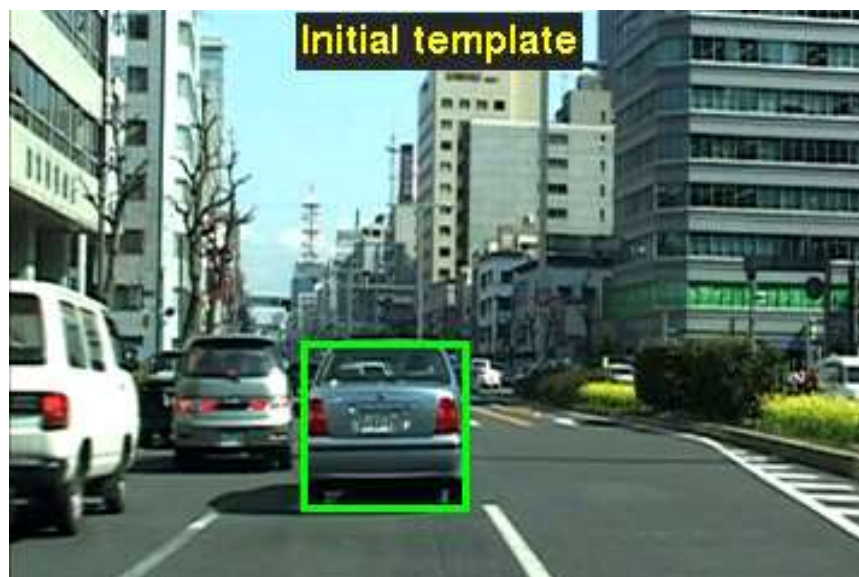
- ◆ 利用光流估计结果求取图像序列中投影点的对应关系
- ◆ 利用光流估计结果求取相邻帧之间物体的运动信息



2. 光流估计的用途—用于稠密运动估计



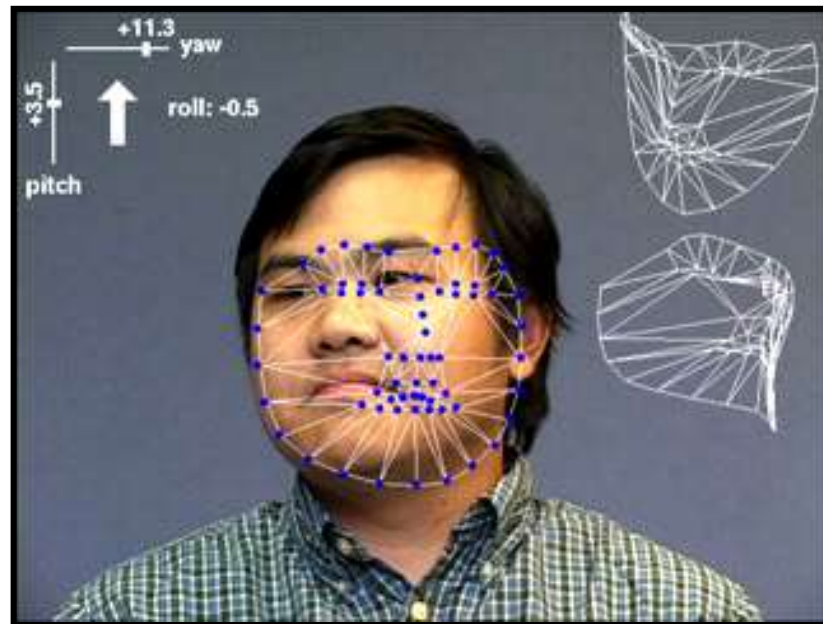
光流估计的用途—刚体运动跟踪



光流估计的用途—非刚体运动跟踪

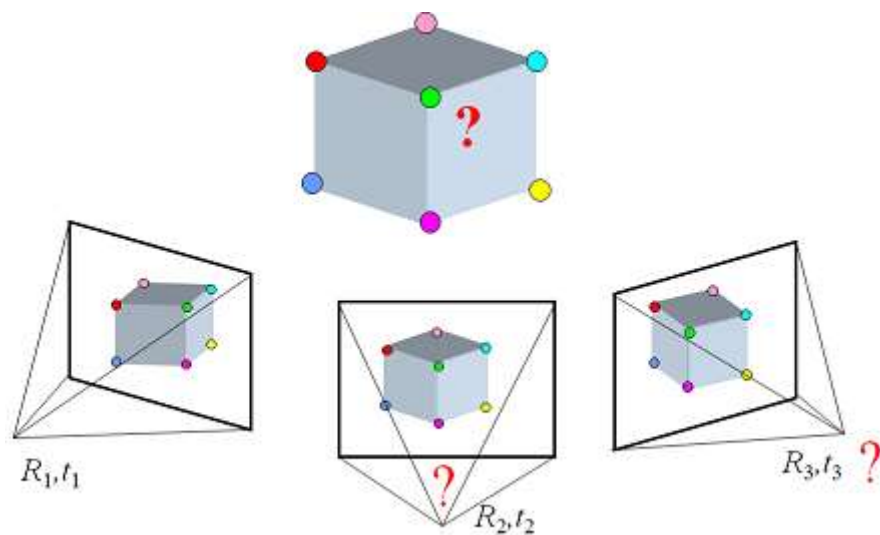


光流估计的用途—人脸跟踪

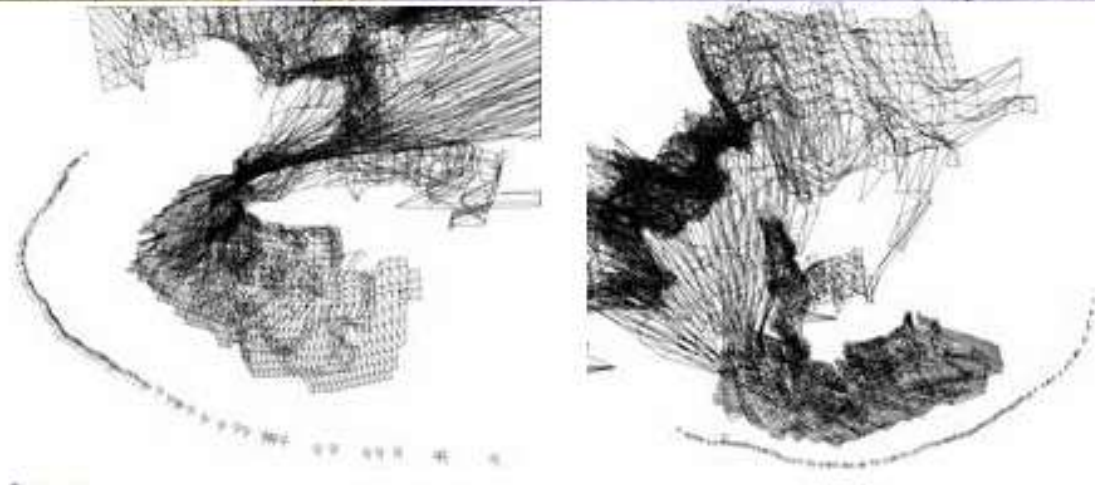


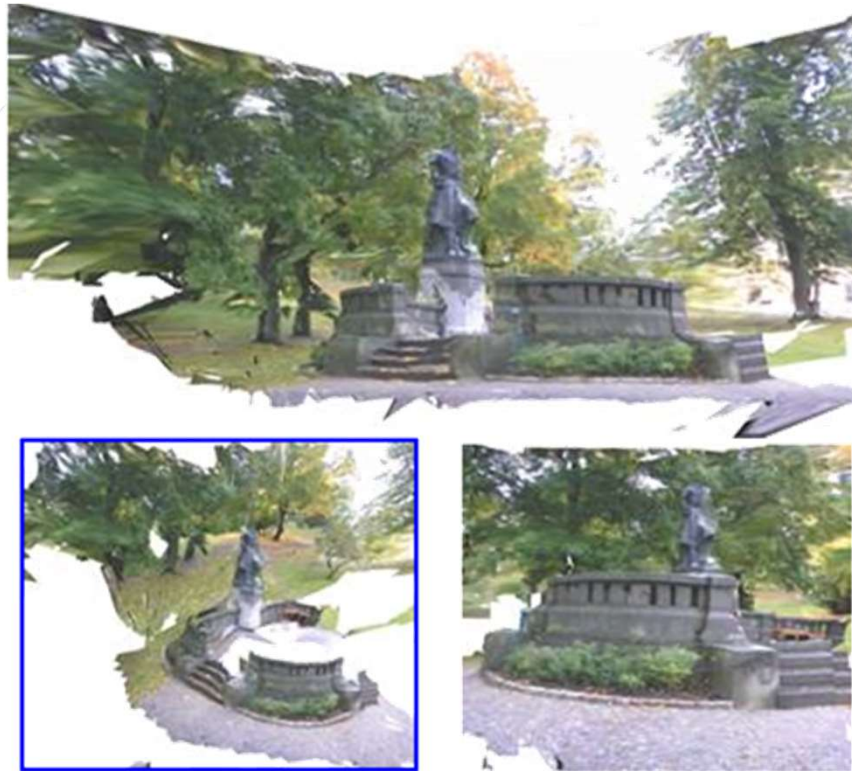
应用于SFM (Structure from Motion)

- ◆ 给定两个或多个图像中的一组对应点，计算相机参数和三维点坐标
- ◆ 给定一组静态场景的图像，其中有对应的二维点
- ◆ 一组3D点 P 和摄像机的旋转 R 和位置 t 来解释观测到的对应关系



应用于SFM





(David Nister, Kentucky)



行为分析的应用



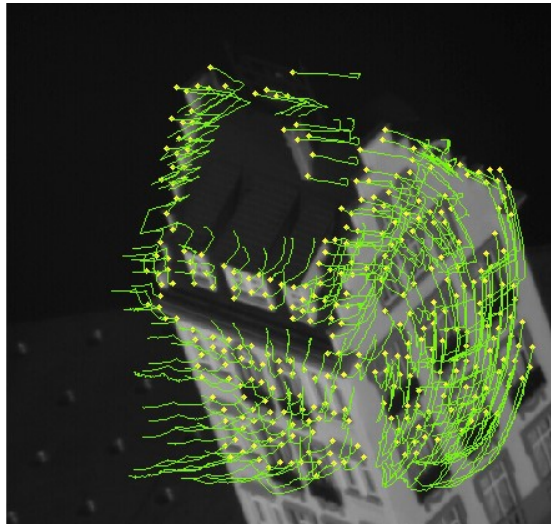
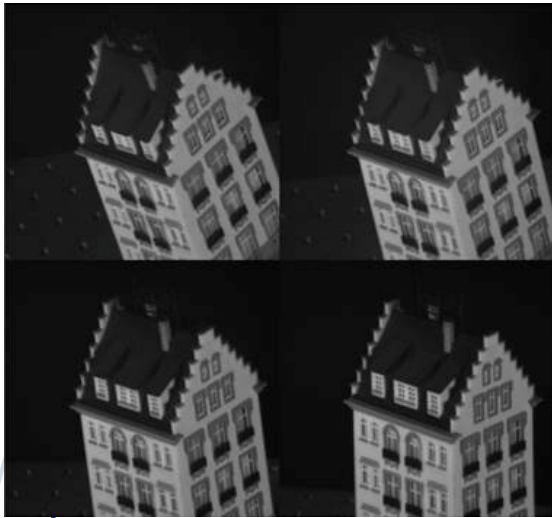
Query



Result



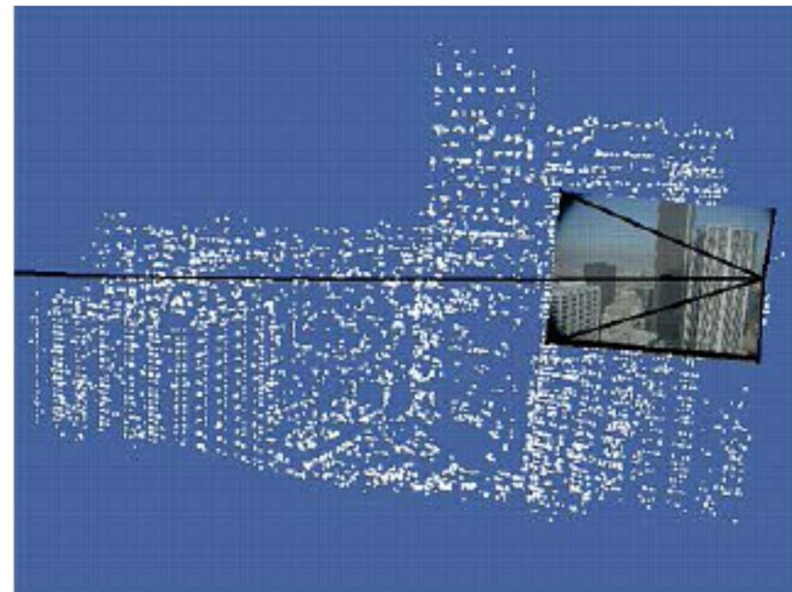
关键点跟踪的应用



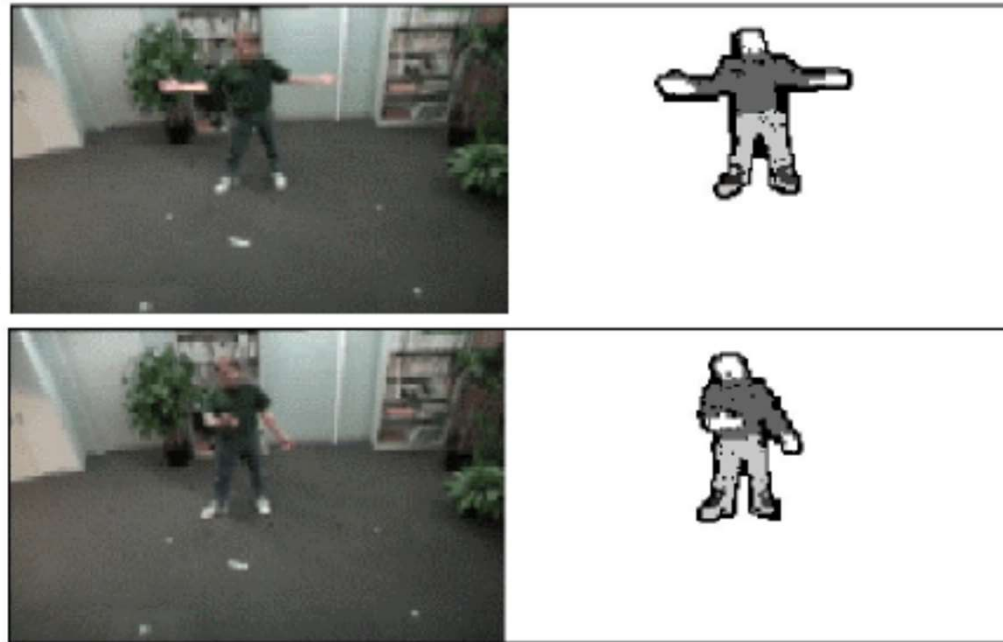
C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. IJCV, 9(2):137-154, November 1992.



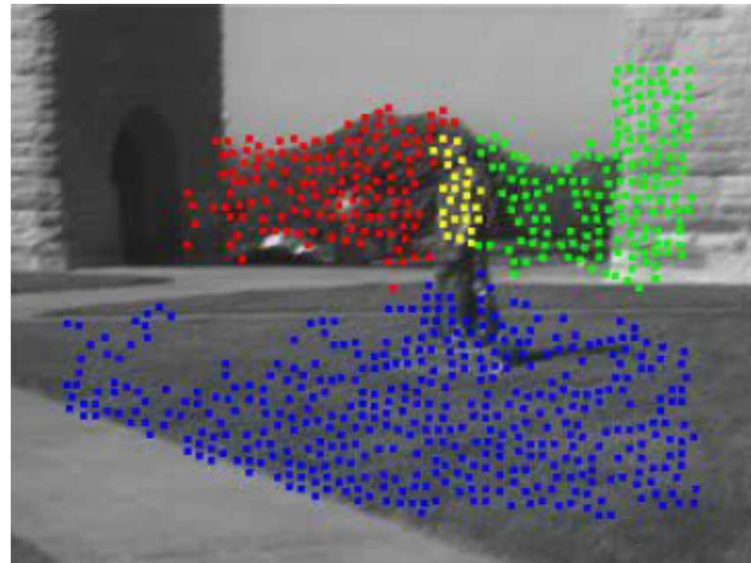
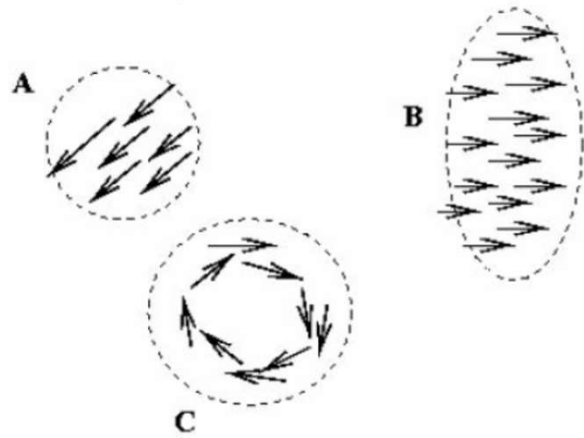
估计场景三维结构



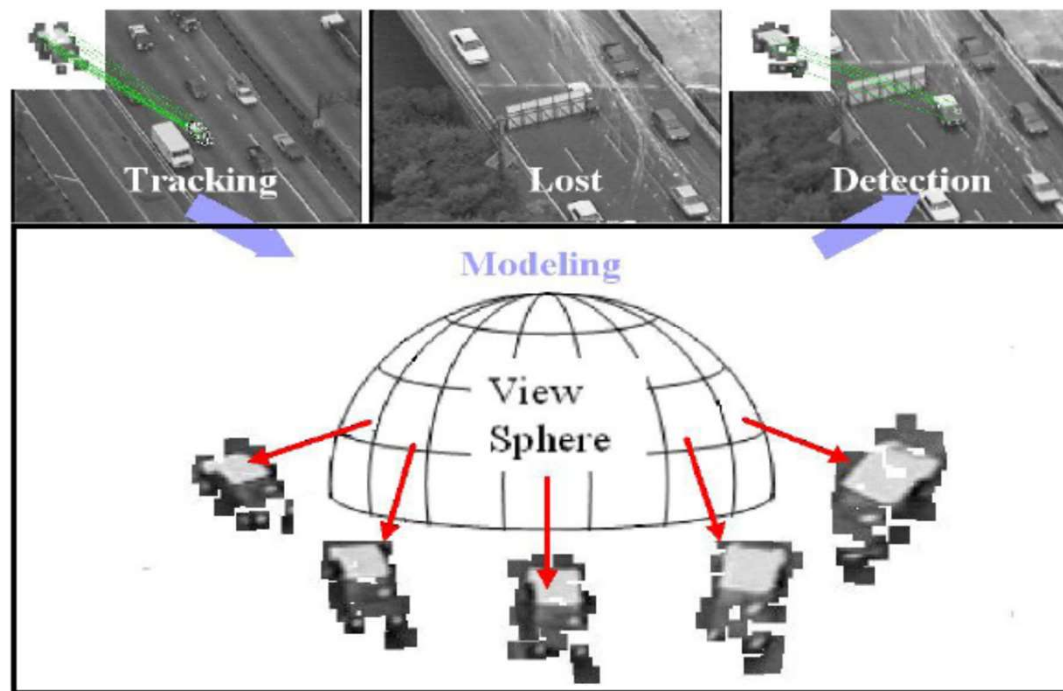
目标分割



基于运动的分割



跟踪目标



Z.Yin and R.Collins, "On-the-fly Object Modeling while Tracking," *IEEE Computer Vision and Pattern Recognition (CVPR '07)*, Minneapolis, MN, June 2007.



合成动态纹理



超分辨率复原

Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn

of the test data o
e of exceptions. T
temperature solder
igated (or some c
facturing technol
etting of 40In40Sn
structural coarse
ycling of 58Bi42Sn

of the test data o
e of exceptions. T
temperature solder
igated (or some c
facturing technol
etting of 40In40Sn
structural coarse
ycling of 58Bi42Sn

of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn

Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn

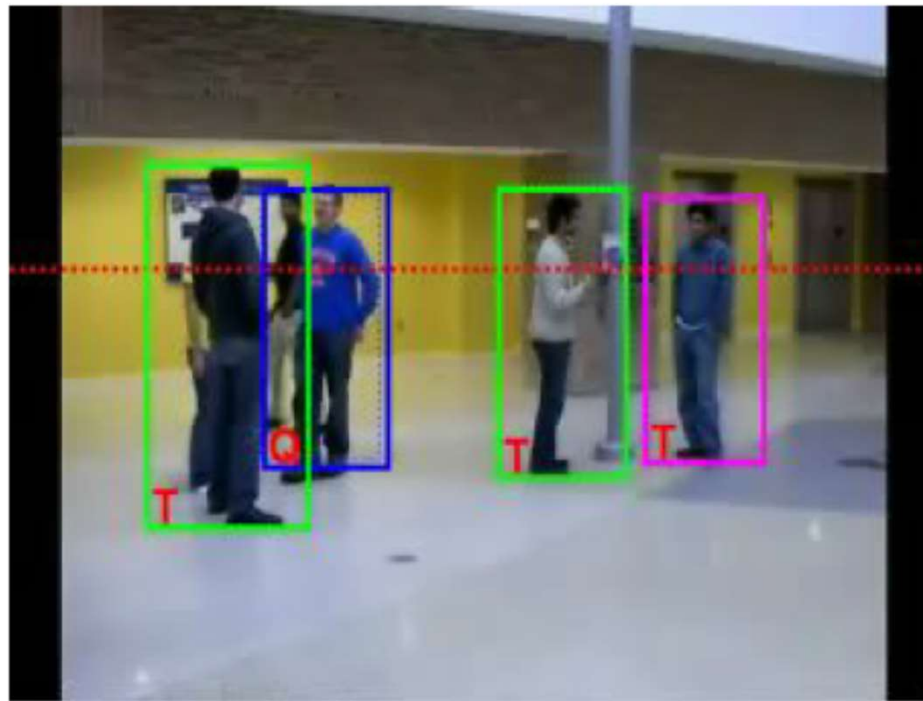
Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn

Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn

Most of the test data o
couple of exceptions. T
low-temperature solder
investigated (or some c
manufacturing technol
nonwetting of 40In40Sn
microstructural coarse
mal cycling of 58Bi42Sn



行为识别



W. Choi & K. Shahid & S. Savarese. [Learning context for collective activity recognition](#), CVPR 2011

Stanford Vision Lab



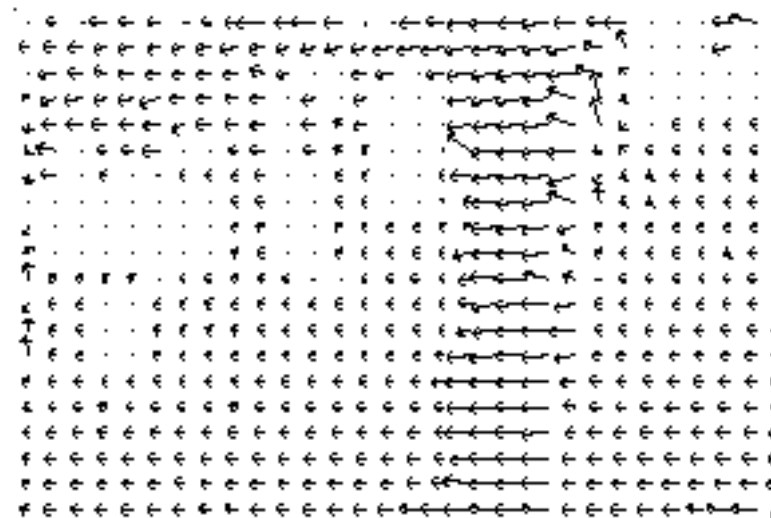
3. 运动场与光流的关系

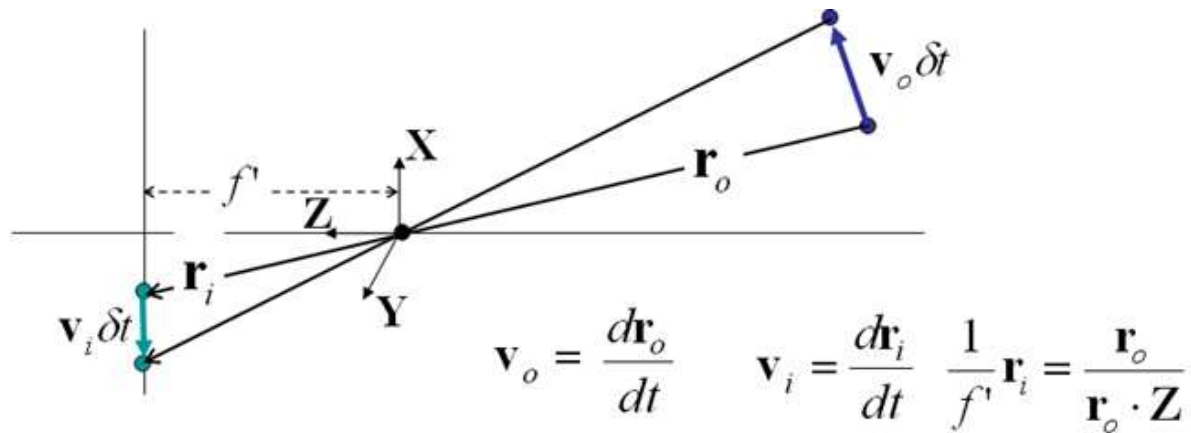
- ◆ 给图像中每一像素点估计得到速度向量，就形成了图像空间2D运动场。
- ◆ 光流法是利用图像序列中像素在相邻帧间的变化来找到相邻帧之间点的相关性，从而计算出相邻帧之间物体的运动信息
- ◆ **光流的产生：**光流是由于场景中前景目标本身的移动、相机的运动，或者两者的共同作用产生的。



光流描述

- ◆ 通常将光流视为一个描述该点瞬时速度的二维矢量 $u=(u, v)$ ，称为光流矢量。
- ◆ 一般情况下，2D成像平面大量点构成的速度场，也称为光流场





Motion field

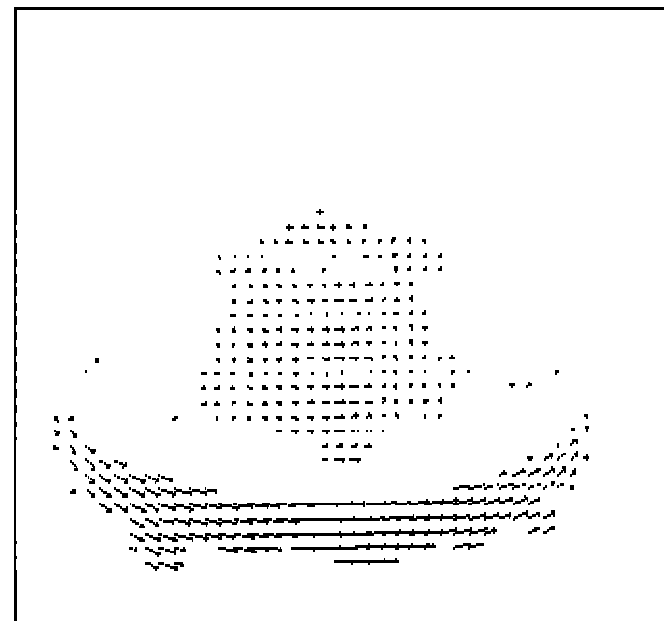
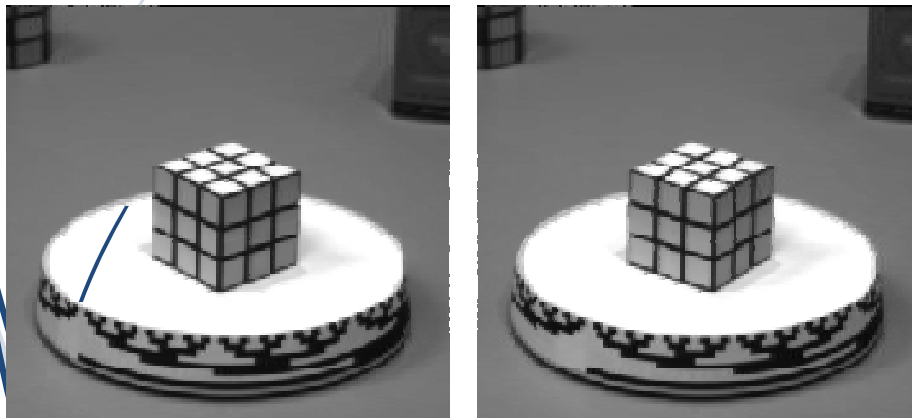
$$\mathbf{v}_i = \frac{d\mathbf{r}_i}{dt} = f' \frac{(\mathbf{r}_o \cdot \mathbf{Z})\mathbf{v}_o - (\mathbf{v}_o \cdot \mathbf{Z})\mathbf{r}_o}{(\mathbf{r}_o \cdot \mathbf{Z})^2} = f' \frac{(\mathbf{r}_o \times \mathbf{v}_o) \times \mathbf{Z}}{(\mathbf{r}_o \cdot \mathbf{Z})^2}$$



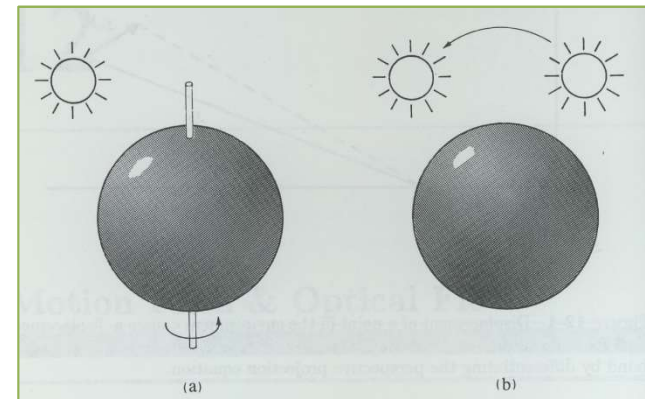
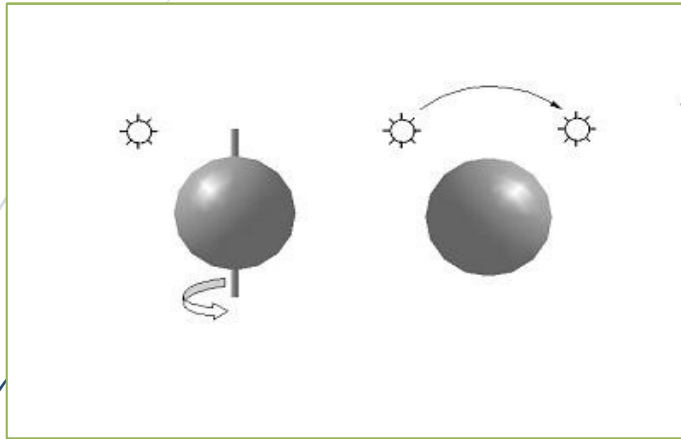
光流场与速度场

◆ 理想情况下:

Optical flow = Motion field

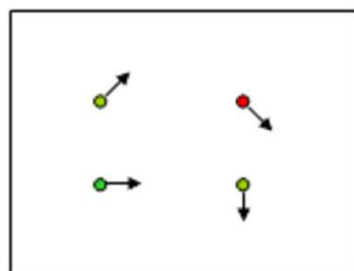


光流场与速度场相等吗？

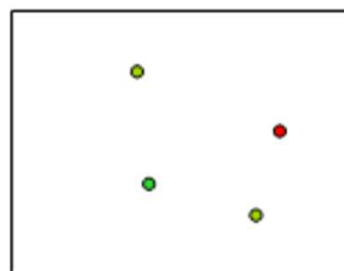


4. 光流的估计方法

- ◆ 给定两个相继帧，估计2D场景任意一点 $P(x, y)$ 的水平速度和垂直速度 (u, v)
其中 u 和 v 与 P 的位置有关，我们记为 $u(x, y)$ 和 $v(x, y)$
- ◆ 光流估计目标： 计算出 u 和 v



$H(x, y)$



$I(x, y)$

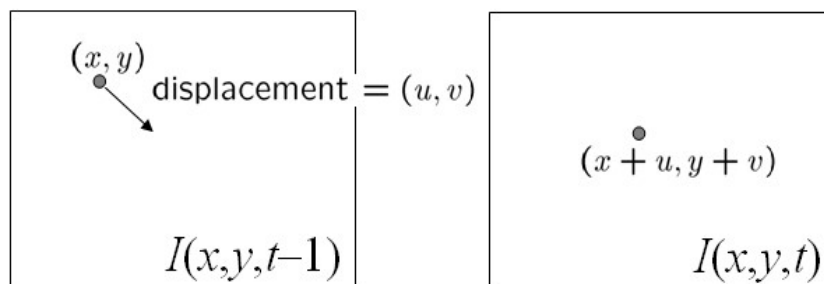
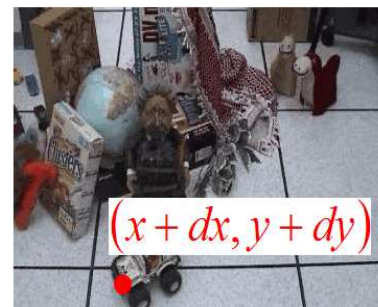


(1) 光流估计的三个重要假设

- ◆ 亮度恒定：每帧中相同点在不同帧中亮度不变
- ◆ 小动作：点的运动位移很小
- ◆ 相干性：对于当前估计的点，其运动与其邻域像素的运动具有相似性



亮度恒定的假设:



◆ 亮度一致性方程

$$I(x, y, t - 1) = I(x + u(x, y), y + v(x, y), t)$$

(2) 光流估计约束

$$I(x, y, t-1) = I(x + u(x, y), y + v(x, y), t)$$

- ◆ 右侧线性化, 用泰勒展开

$$I(x, y, t-1) \approx I(x, y, t) + I_x u(x, y) + I_y v(x, y)$$

- ◆ 亮度恒定, 得到: $I_x u + I_y v + I_t \approx 0$

$$\text{即 } \nabla I \cdot (u, v) + I_t = 0$$

I_x, I_y, I_t 利用梯度计算

物理意义: 如果一个固定的观察者观察一幅活动的场景, 那么所得图象上某点灰度的变化率(时间一阶) 是场景亮度变化率与该点运动速度的乘积。

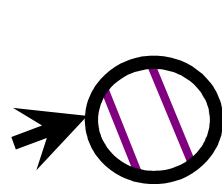


光流估计

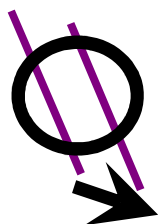
- ◆ 光流估计目标： 计算出 u 和 v
- ◆ 出现的问题：图像中的每一点上有两个未知数 u 和 v ，但只有一个方程，因此，只是利用一个像素点信息不能确定光流 (u, v) .
 - 最优化问题, 迭代求解
 - 增加约束，使问题可解



(3) 光流估计中的孔径问题 (Aperture problem)



感知运动

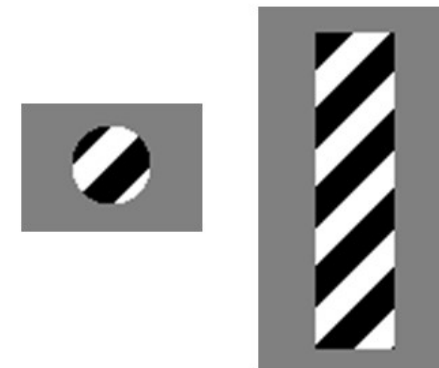
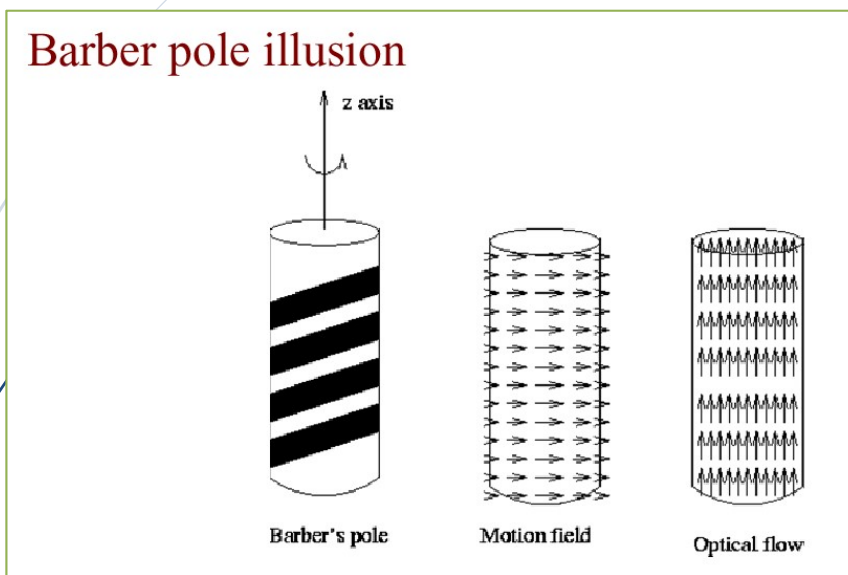


实际运动

- ◆ 结果： 有运动，感知不到光流
- ◆ 根本原因： 光流估计需要足够的图像梯度信息，才能进行准确的估计
 - 对于无纹理区域或者沿着边缘方向无法使用
 - 只能在角点和纹理区域使用



理发杆幻觉现象 (The barber pole illusion)



(4) 光流估计

- ◆ 光流约束中的误差公式:

$$e_c = \iint_{image} (I_x u + I_y v + I_t)^2 dx dy$$

- ◆ 需要附加额外约束

- ◆ 平滑项约束:

$$e_s = \iint_{image} (u_x^2 + u_y^2) + (v_x^2 + v_y^2) dx dy$$



(4) 光流估计常用方法

- ◆ 基于梯度的方法
 - Horn-Schunck法
 - Lucas-Kanade方法
 - Nagel方法
- ◆ 块匹配方法



(a) 基于梯度方法 — Horn-Schunck法

第一阶段：连续求解方法

◆ Step1: 根据光流约束方程, 光流误差为 $e^2(\mathbf{x}) = (I_x u + I_y v + I_t)^2$ $\mathbf{x} = (x, y)^T$

◆ Step2: 根据光滑变化的假设, 其速度分量平方和积分为

$$s^2(\mathbf{x}) = \iint \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] dx dy$$

◆ Step3: 将光滑性与微分约束加权组合: $E = \iint \{ e^2(\mathbf{x}) + \alpha s^2(\mathbf{x}) \} dx dy$

◆ 其中 α 是控制平滑度的参数, 其越大, 则平滑度就越高, 使用变分法将上式转化为一对偏微分方程: $\alpha \nabla^2 u = I_x^2 u + I_x I_y v + I_x I_t$

$$\alpha \nabla^2 v = I_x I_y u + I_y^2 v + I_y I_t$$

◆ Step4: 用有限差分方法将每个方程中拉普拉斯算子换成局部邻域图像光流矢量的加权和, 并使用迭代方法求解step3中两个差分项。



Horn-Schunck法

◆ 第二阶段：离散化迭代求解

- 离散的情况下，根据光流约束方程，对光流误差进行离散表示，利用点 (i, j)

及其4邻域光流值差分及光流约束方程计算，极小化函数 $E = \sum_i \sum_j (e^2(i, j) + \alpha s^2(i, j))$

- 得到光流的迭代计算公式：
$$u^{n+1} = \bar{u}^n - I_x \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{\alpha + I_x^2 + I_y^2}$$
$$v^{n+1} = \bar{v}^n - I_y \frac{I_x \bar{u}^n + I_y \bar{v}^n + I_t}{\alpha + I_x^2 + I_y^2}$$

其中 \bar{u} 和 \bar{v} 分别是 u 和 v 在点 (i, j) 处的平均值



Lucas-Kanade方法

- ◆ 一种广泛使用的光流估计的差分方法，由Bruce D. Lucas和Takeo Kanade发明。整个图像上光滑变化的假设来求解光流，即运动场既满足光流约束方程又满足全局平滑性，其假设光流在像素点的邻域是一个常数，然后通过最小二乘法利用邻域像素点，对光流方程求解。
- ◆ L-K方法结合邻近像素点信息，消除光流方程的多义性。而且，与逐点计算的方法相比，L-K方法对图像噪声不敏感，一种空域局部方法，所以在图像的均匀区域内部，L-K方法无法得到光流信息。
- ◆ 恒定亮度是指某些像素的跟踪不随时间变化：

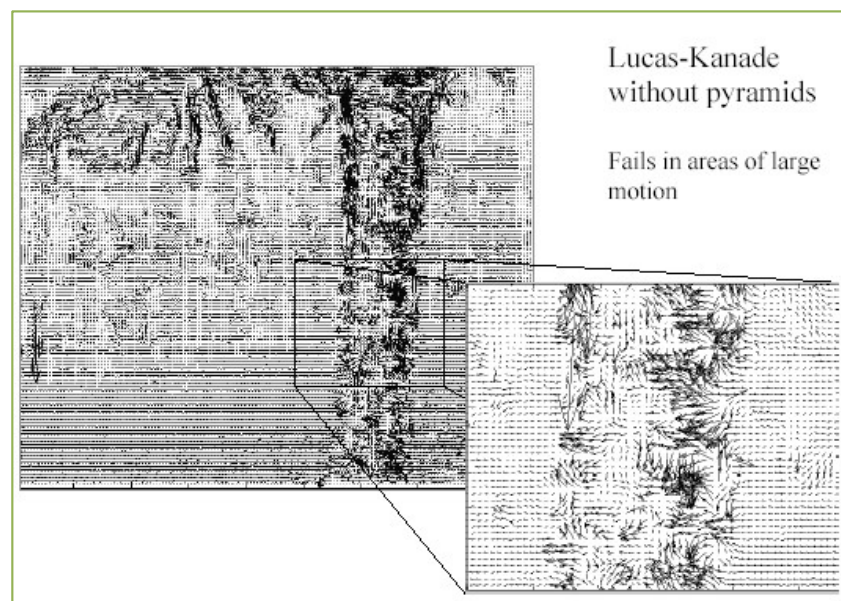


Lucas-Kanade方法

- ◆ 主要思想： Lucas和Kanade [Lucas 1981] 假设在一个小的空间邻域 上运动矢量保持恒定，然后使用加权最小二乘法估计光流。在一个小的空间邻域上，对光流估计误差进行优化处理：

$$\sum_{(x,y) \in \Omega} W^2(\mathbf{x})(I_x u + I_y v + I_t)^2$$

其中 $W(\mathbf{x})$ 表示窗口权重函数



传统L-K方法存在的问题

- ◆ 因为lk算法的约束条件即：小速度，亮度不变以及区域一致性都是较强的假设，并不很容易得到满足。
- ◆ 如当物体运动速度较快时，假设不成立，那么后续的假设就会有较大的偏差，使得最终求出的光流值有较大的误差。
- ◆ 为了克服物体运动速度较大时，算法出现较大误差问题，减少图像中物体的运动速度，采取措施：缩小图像的尺寸。
- ◆ 改进措施：光流通过生成原图像的金字塔图像，逐层求解，不断精确来求得。



Lucas-Kanade改进算法

◆ 主要的步骤有三步：

- 建立金字塔
- 金字塔跟踪
- 迭代过程



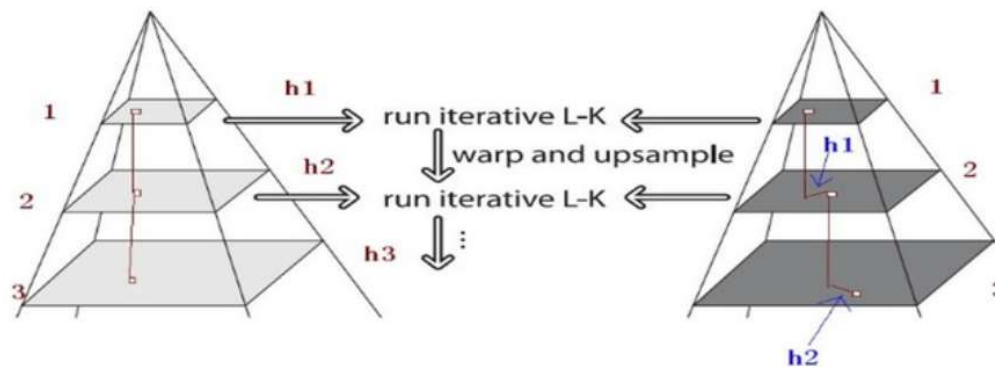
建立金字塔

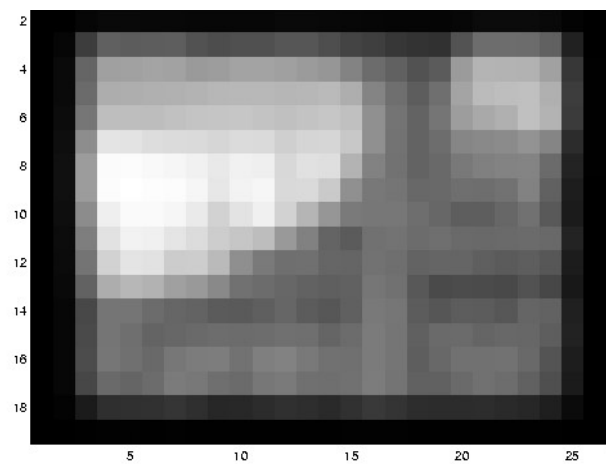
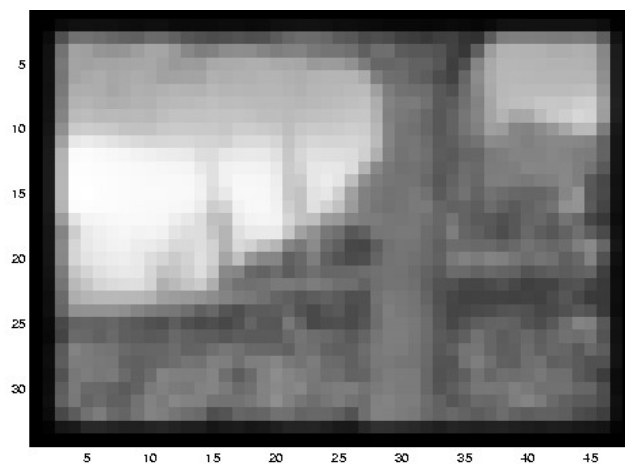
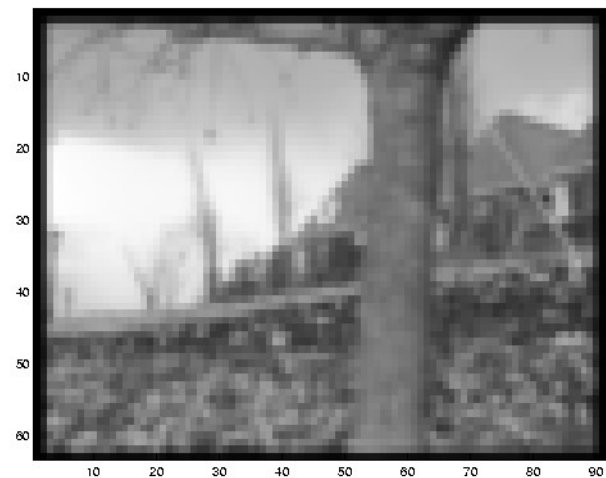
- ◆ 用一个 $[0.25 \ 0.5 \ 0.25]$ 的低通滤波器对 I_{L-1} 进行卷积

$$I^L = \frac{1}{4} I^{L-1}(2x, 2y) +$$

$$\frac{1}{8} (I^{L-1}(2x-1, 2y) + I^{L-1}(2x+1, 2y) + I^{L-1}(2x, 2y-1) + I^{L-1}(2x, 2y+1)) +$$

$$\frac{1}{16} (I^{L-1}(2x-1, 2y-1) + I^{L-1}(2x+1, 2y-1) + I^{L-1}(2x-1, 2y+1) + I^{L-1}(2x+1, 2y+1))$$





金字塔跟踪

- ◆ 首先，光流和仿射变换矩阵在最高一层的图像上计算出；
- ◆ 将上一层的计算结果传递给当前层作为初始值，进一步在这个初始值的基础上，计算当前层的光流和仿射变化矩阵；
- ◆ 再将当前层的光流和仿射矩阵作为初始值传递给下一层图像，
- ◆ 直到传递给最后一层，即原始图像层，计算出来的光流和仿射变换矩阵作为最后的光流和仿射变换矩阵的结果。



迭代过程

- ◆ 在金字塔的每一层，计算出光流 dL 和仿射变换矩阵 AL ，目标使误差 ξL 最小
- ◆ 方法： 将上一层的光流 u 传给当前层，计算当前层像素强度，计算出图像在该点 x 方向和 y 方向上的偏导

$$I(x) \leftarrow I^L(x + u / 2^L)$$

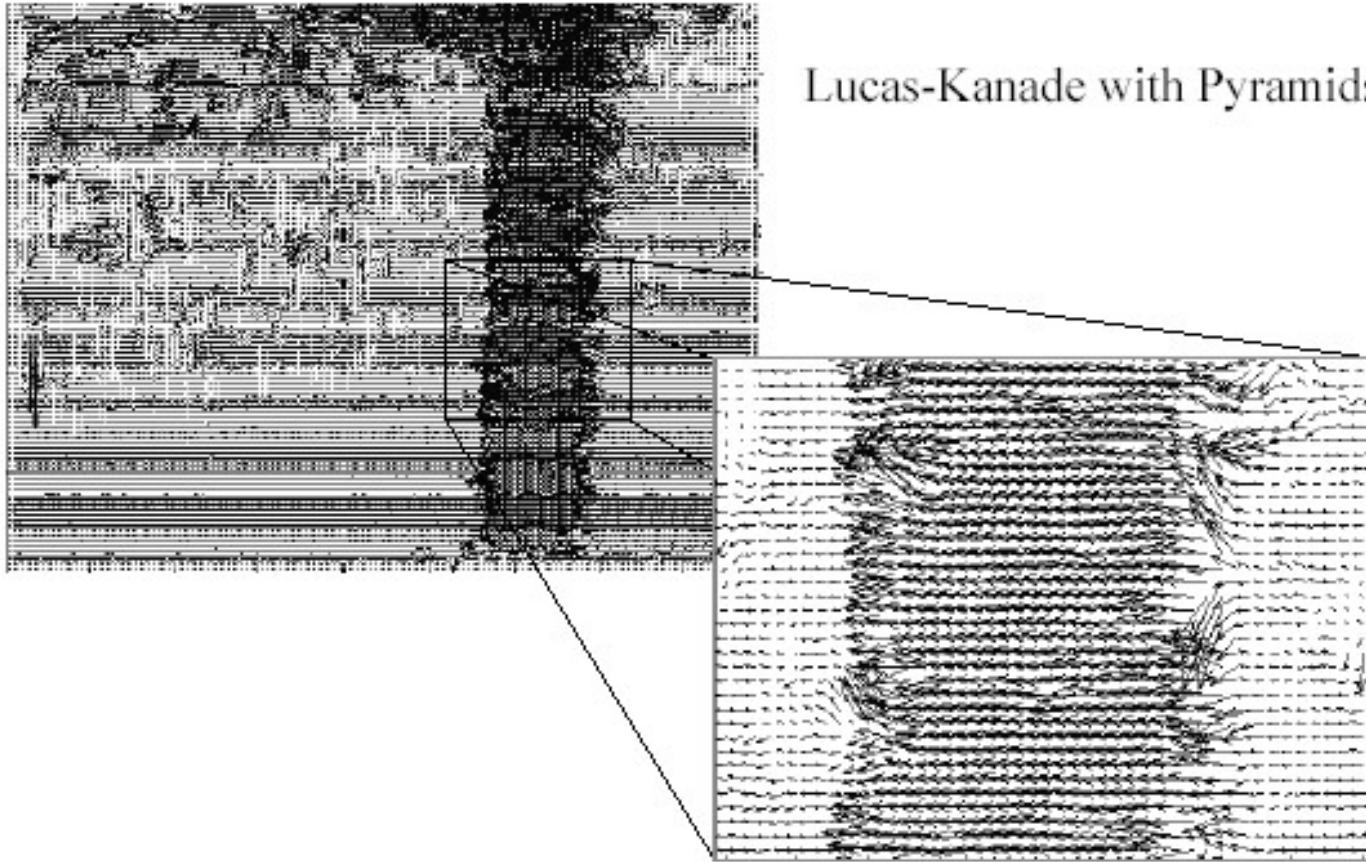
$$I_x = [I(x+1, y) - I(x-1, y)] / 2$$

$$I_y = [I(x, y+1) - I(x, y-1)] / 2$$

- ◆ 根据两帧图像间相同位置点的灰度值之差，更新跟踪结果，不断迭代



Lucas-Kanade with Pyramids



Nagel 方法

- ◆ Nagel 使用二阶导数 (second-order derivatives) 来估计光流。和 Horn-Schunck 法一样，Nagel 也使用了全局平滑约束来建立光流误差测度函数。
- ◆ 与 Horn-Schunck 不同，Nagel 提出的一种面向平滑的约束，并不是强加在亮度梯度变化最剧烈的方向（即边缘方向）上，是为了处理遮挡 (occlusion) 问题。该方法的误差测度函数为

$$\iint (I_x u + I_y v + I_t)^2 + \frac{\alpha^2}{\|\nabla I\|_2^2 + 2\delta} [(u_x I_y - u_y I_x)^2 + (v_x I_y - v_y I_x)^2 + \delta(u_x^2 + u_y^2 + v_x^2 + v_y^2)] dx dy$$

Nagle 方法

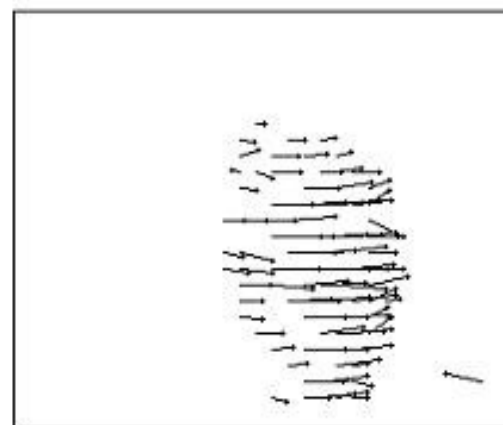
- ◆ 基于二阶导数的方法
- ◆ 面向平滑的约束，处理遮挡
- ◆ Gauss-Seidel 迭代求解

$$u^{k+1} = \xi(u^k) - \frac{I_x[I_x \xi(u^k) + I_y \xi(v^k) + I_t]}{I_x^2 + I_y^2 + \alpha^2}$$

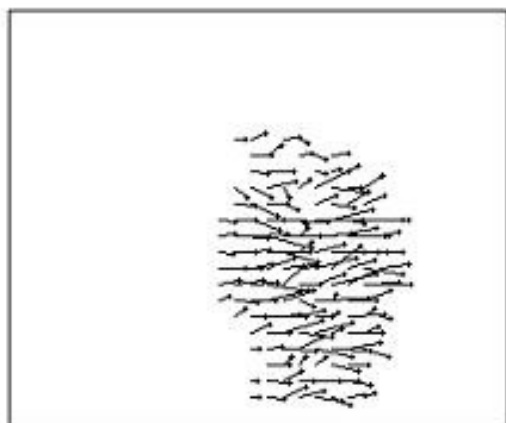
$$v^{k+1} = \xi(v^k) - \frac{I_y[I_x \xi(u^k) + I_y \xi(v^k) + I_t]}{I_x^2 + I_y^2 + \alpha^2}$$



Horn-Schunck



Lucas-Kanade



Nagel



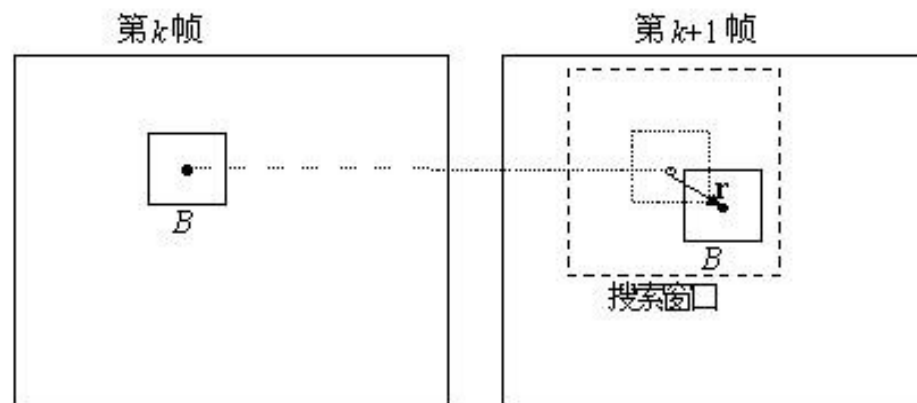
第20帧

梯度方法的不足

- ◆ 当窗口内图像结构信息不明确时算法失效
- ◆ 当位移很大时算法失效(一般而言有效距离是1个像素)
 - 亮度的线性分解只在小位移条件下成立
- ◆ 而且图像的亮度也并不是严格意义下的一致
 - 通常需要对图像进行预处理(平滑滤波),使得图像看起来更加相似

(b) 块匹配方法

- ◆ 将图像帧划分为块
- ◆ 对于当前帧中的图像块，搜索其在下一帧中的对应块
- ◆ 计算对应块位移矢量



◆ 块匹配算法的基本思想：在第 i 帧中选择一个块及中心，然后在第 $i+1$ 帧中的一个较大的搜索窗口内寻找与预选块尺寸相同的最佳匹配块的中心的位移矢量。搜索窗口一般是以第 i 帧中块预选中心为基准的一个对称窗口，其大小常常根据先验知识或经验来确定。各种块匹配算法的差异主要体现在如下几个方面：

- 匹配准则
- 搜索策略
- 块尺寸选择方法



块匹配准则

1. 最小均方差 (MSE)

$$MSE(\Delta x, \Delta y) = \frac{1}{mn} \sum_{(x,y) \in W} [I(x, y, k) - I(x + \Delta x, y + \Delta y, k + 1)]^2$$

2. 最小平均绝对差 (MAD)

$$MAD(\Delta x, \Delta y) = \frac{1}{mn} \sum_{(x,y) \in W} |I(x, y, k) - I(x + \Delta x, y + \Delta y, k + 1)|$$

3. 最大匹配像素 (MPC)

$$MPC(\Delta x, \Delta y) = \sum_{(x,y) \in W} T(x, y, \Delta x, \Delta y) \quad [\Delta x, \Delta y]^T = \arg \min_{(\Delta x, \Delta y)} MPC(\Delta x, \Delta y)$$

$$T(x, y, \Delta x, \Delta y) = \begin{cases} 1 & \text{如果 } |I(x, y, k) - I(x + \Delta x, y + \Delta y, k + 1)| \leq T \\ 0 & \text{其它} \end{cases}$$

搜索策略

- ◆ 全搜索策略：为了求得最佳位移估计，可以计算所有可能的位移矢量对应的匹配误差，然后选择最小匹配误差对应的矢量就是最佳位移估计值
- ◆ 优点：可以找到全局最优值，但十分浪费时间，实际中经常采用快速搜索策略

5. 密集光流（稠密光流）与稀疏光流

- ◆ **稠密光流：** 是一种针对图像进行逐点匹配的图像配准方法 。
- ◆ **稀疏光流：** 针对图像上若干个特征点。
- ◆ **区别：** 稠密光流计算图像上所有的点的偏移量，从而形成一个稠密的光流场。通过
这个稠密的光流场，可以进行像素级别的图像配准，所以其配准后的效果也明显优于
稀疏光流配准的效果。但是存在计算量大的问题。

从稀疏光流到稠密光流

- ◆ 局部加权线性回归插值稠密光流场
- ◆ 稀疏特征点聚类的流向量



OpenCV计算光流的函数

- 1) `calcOpticalFlowPyrLK` : 通过金字塔Lucas-Kanade 光流方法计算某些点集的光流（稀疏光流）。
- 2) `calcOpticalFlowFarneback`: 用Gunnar Farneback 的算法计算稠密光流（即图像上所有像素点的光流都计算出来）
- 3) `CalcOpticalFlowBM`: 通过块匹配的方法来计算光流。
- 4) `CalcOpticalFlowHS`: 用Horn-Schunck算法计算稠密光流。
- 5) `calcOpticalFlowSF`: 非递归的优化方法。