

## 第12章 运动跟踪



计算机科学与技术学院

# 本次课程内容

1. 目标跟踪问题

2. 目标跟踪的应用

3. 目标识别问题难点

4. 动态跟踪的物体表达

5. 动态跟踪流程框架

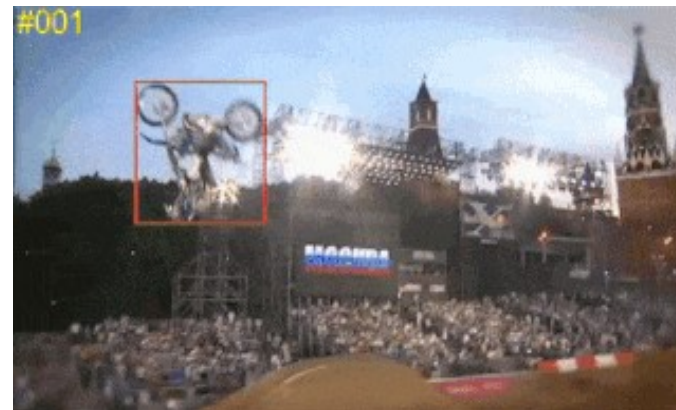
6. 视觉跟踪算法

7. 深度学习的跟踪方法



## 1. 目标跟踪问题

- ◆ **目标跟踪：** 根据视频初始帧中目标大小与位置，预测、寻求后续帧中的目标并生成连续的轨迹。
- ◆ **目的：** 估计目标在跨帧后的状态
- ◆ **状态：** 位置、外观、形状、速度等

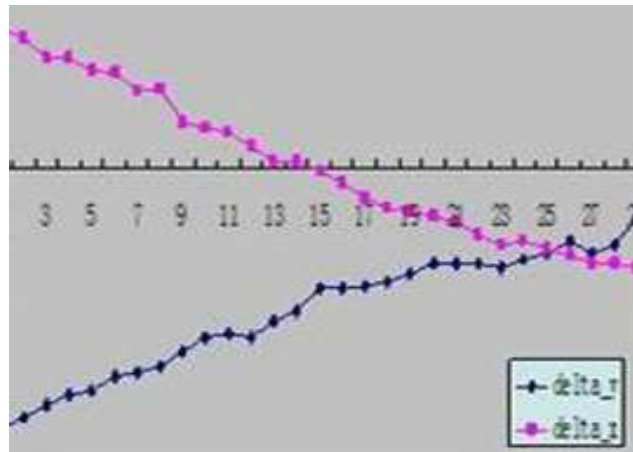


## 跟踪的目标

- ◆ **目标：**图像中明显区别于周围环境的闭合区域，具有一定的物理意义。



## 跟踪的结果：序列位置的轨迹



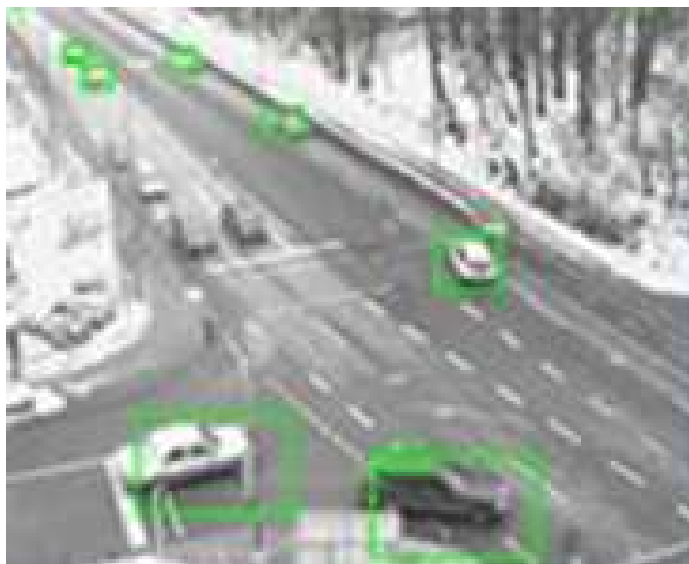
## 目标的检测

- ◆ 目标跟踪过程一般需要检测目的检测
- ◆ 目标检测：给出目标在图像位置。

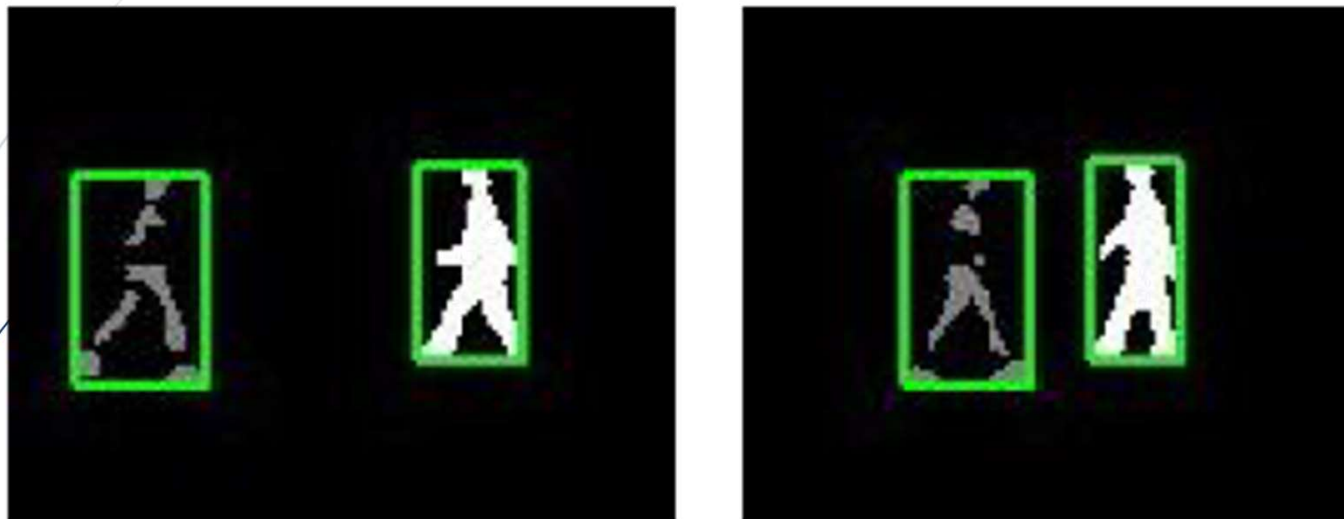


## 单目标跟踪与多目标跟踪

- ◆ 目标跟踪分为单目标跟踪和多目标跟踪



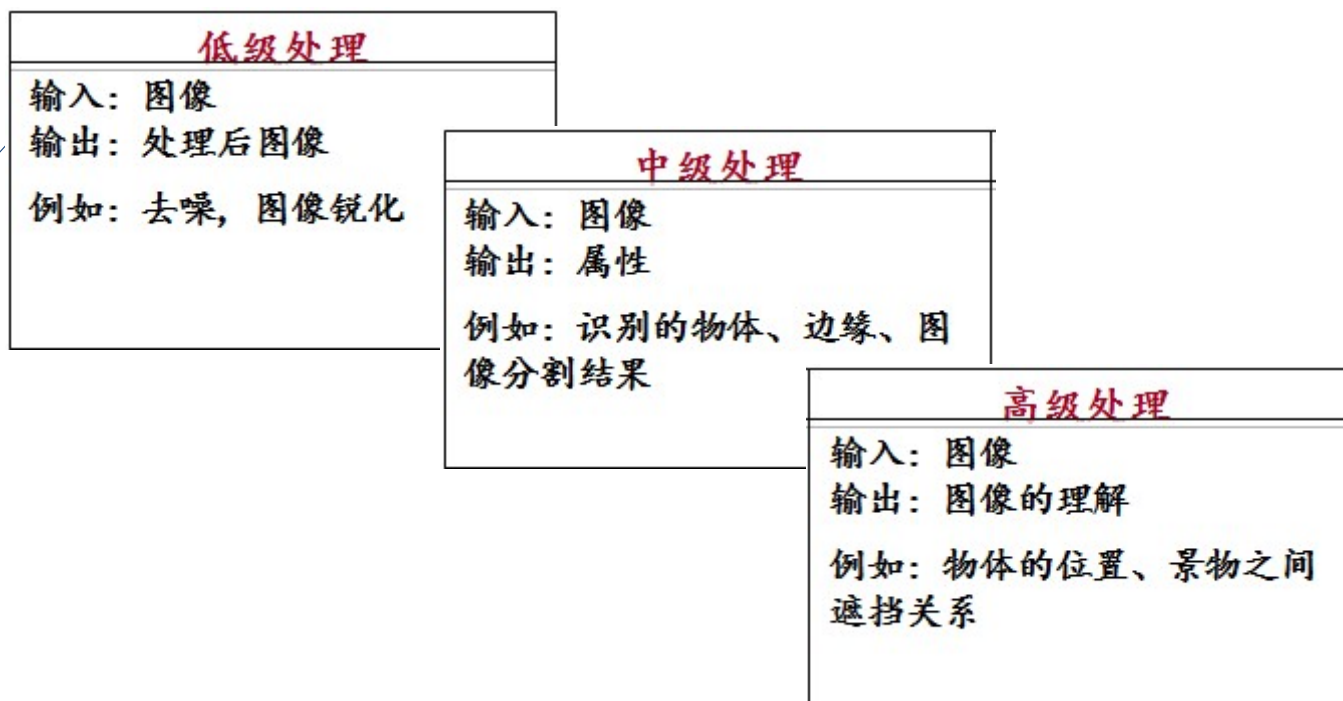
## 多目标跟踪实例





## 2. 目标跟踪问题的应用

- ◆ 在计算机视觉的三层结构中，目标跟踪属于中间层，是其他的高层任务（如动作识别，行为分析等）的基础。



## 2. 目标跟踪的应用

### ◆ 目标跟踪的主要应用包括：

- 视频监控。检测异常行为，节省大量的人力物力。
- 人机交互。对于复杂场景中目标交互的识别与处理。
- 虚拟现实和增强现实。比如游戏中角色运动的设置等。
- 医学图像。



## 应用领域1—可视化跟踪



## 应用领域2—视频监控



## 应用领域3—行人重识别



(a). Generated Images



(b). Original Images

### 3. 目标识别问题难点

- ◆ 几何变化引起的目标形状在不同帧中的变化（形变物体）
  - 姿势、红旗、规模（尺度）、海水目标
- ◆ 光度因素引起目标在不同帧中的变化
  - 照明、外观
- ◆ 目标遮挡时，跟踪目标成为难点



## 变形物体跟踪较难





## 场景光强发生明显变化

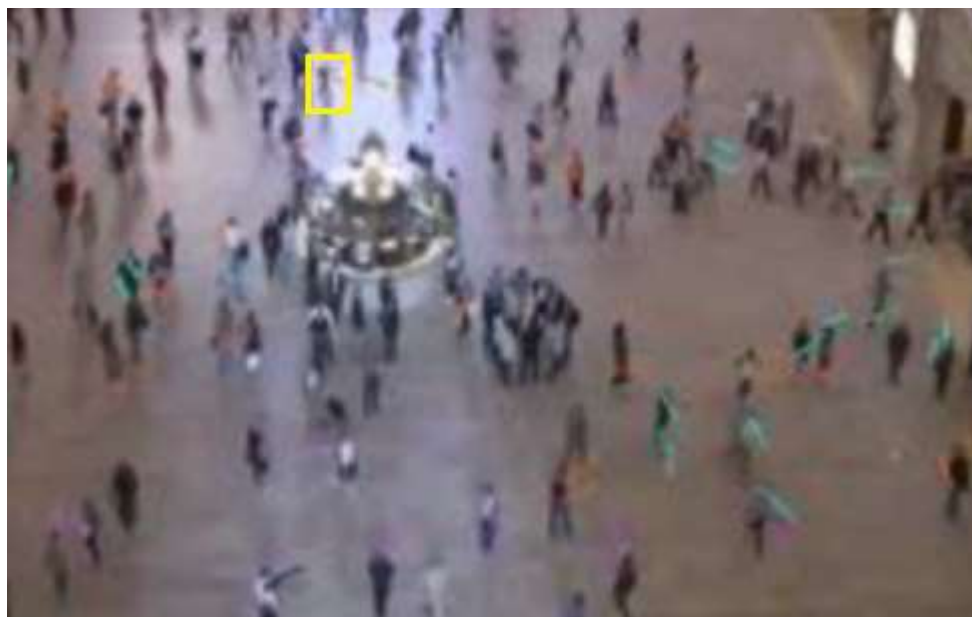


illumination variation

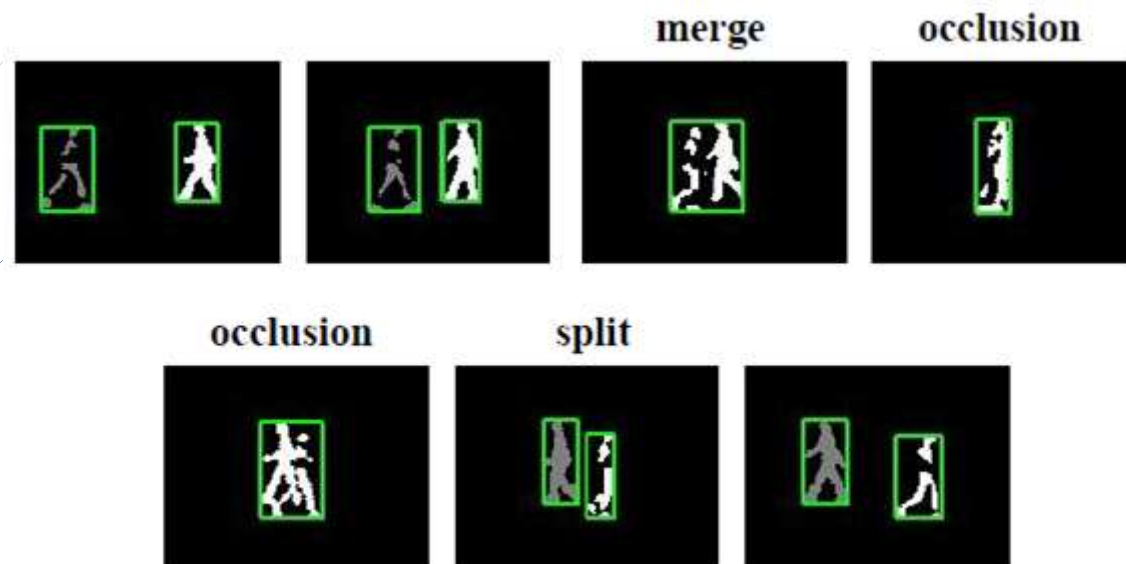




## 小目标跟踪较难



## 遮挡或者分离的目标跟踪较难



## 4. 动态跟踪的物体表达

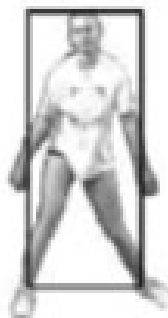
◆ 目标表达： 足以能将目标从背景中分离出来

◆ 物体近似表达方法：

➤ 分割出区域/多边形



➤ 椭圆形包围盒/矩形包围盒



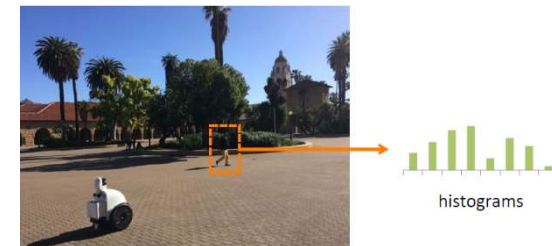
➤ 仅仅表示位置



## 跟踪物体的表达

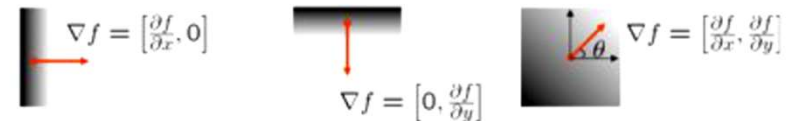
### ◆ 低级（层）/中低级（层）/高（层）特征

#### ➤ 低级（层）特征：边缘梯度



### Image gradient

- The gradient of an image:  $\nabla f = \left[ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$



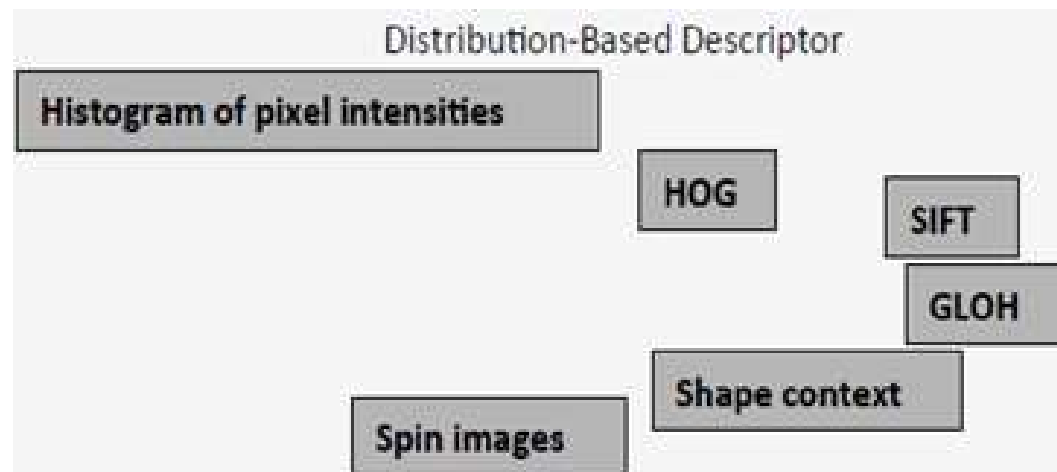
points in the direction of most rapid increase in intensity

direction is given by  $\theta = \tan^{-1} \left( \frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$

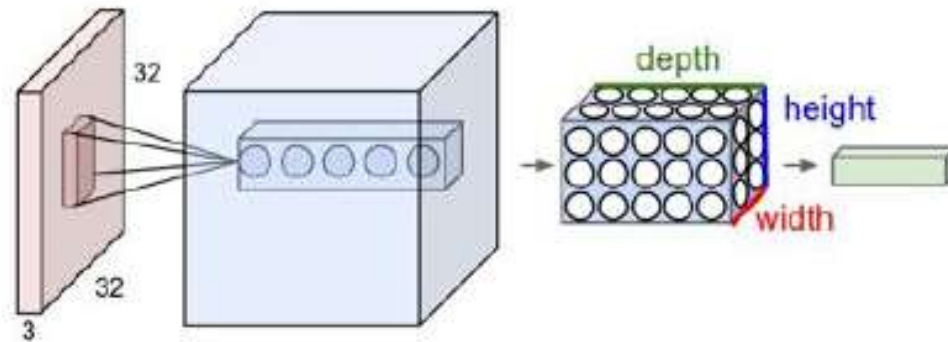
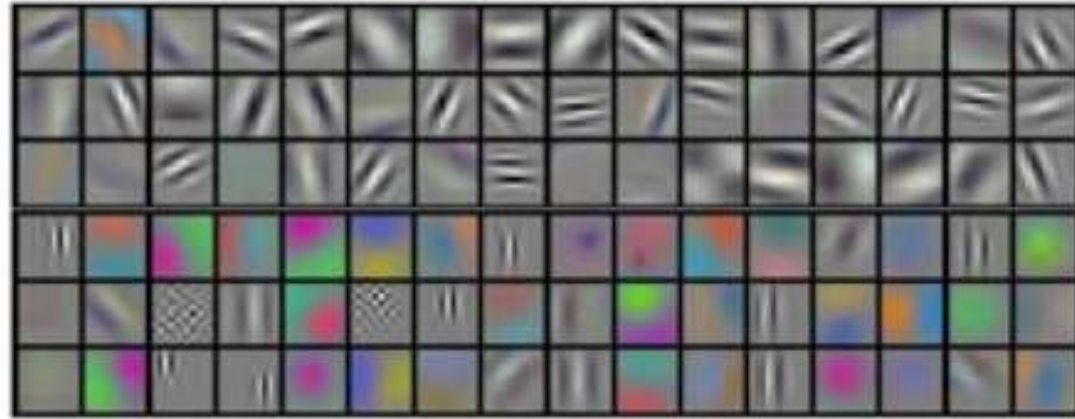
does this relate to the direction of the edge?



## 低级（层）特征

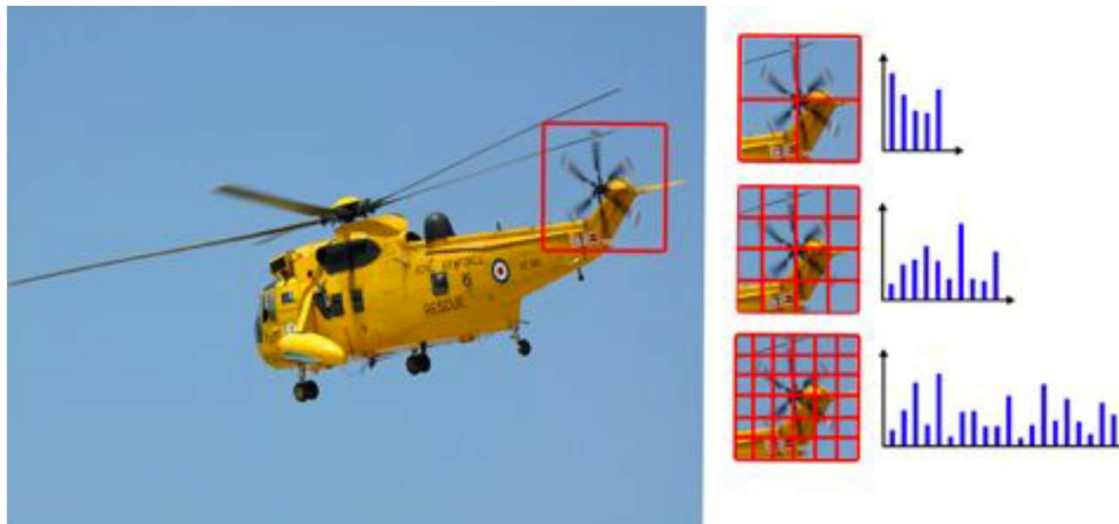


## CNN features



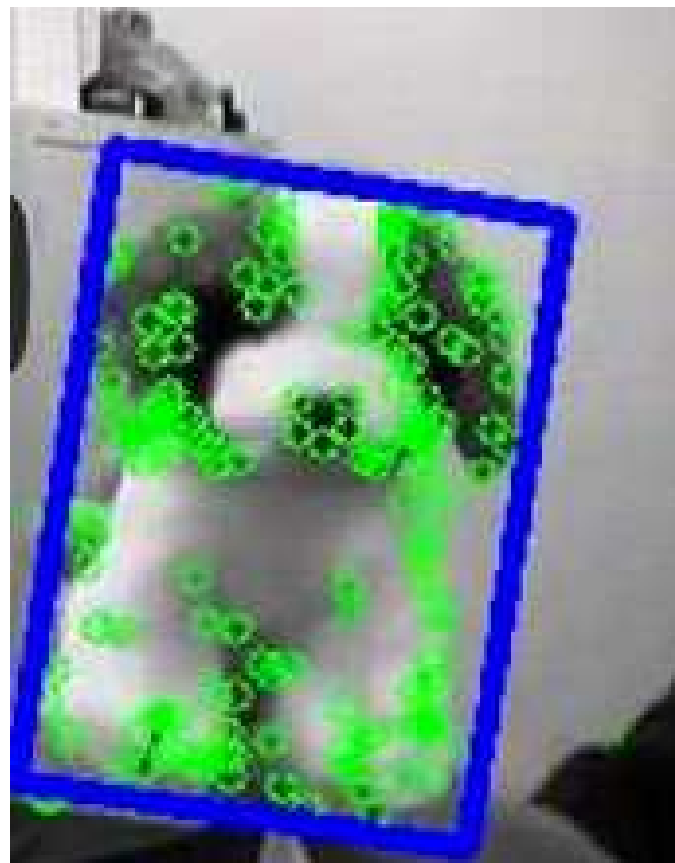
## 跟踪物体的表达—采样

- ◆ 多层采样: pyramid/cascade of coarse-to-fine



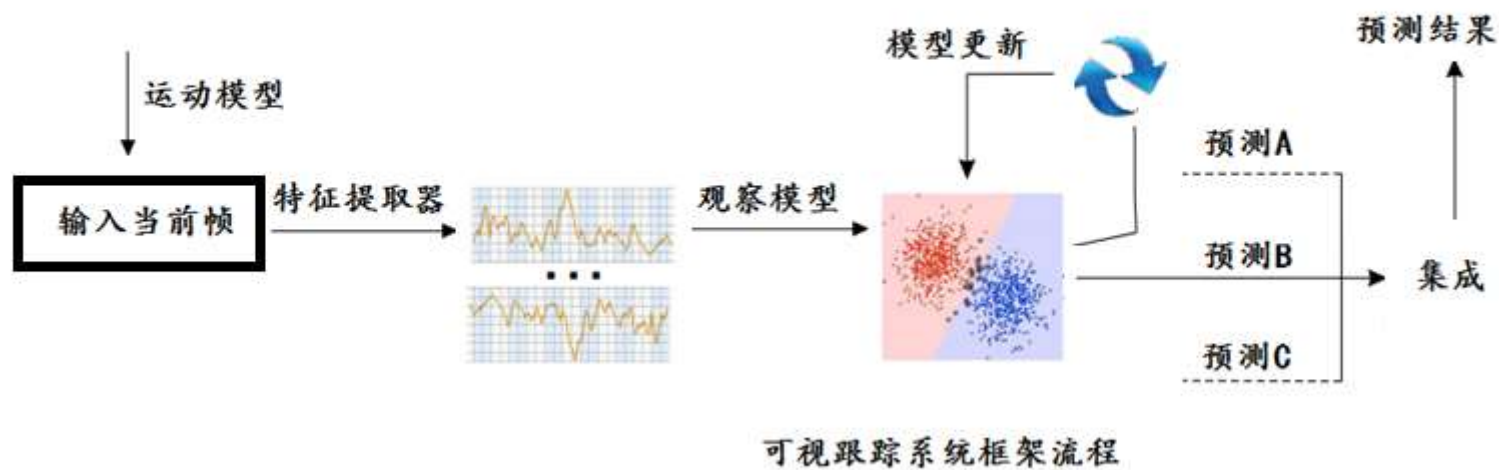
## 跟踪物体的表达—采样

- ◆ 局部Patch及采样点（特征点）





## 5. 动态跟踪流程框架



## 5. 动态跟踪流程一步骤1

- ◆ 输入初始化目标框
- ◆ 建立运动模型
- ◆ 然后，在下一帧中产生多个候选框



## 动态跟踪流程一步骤2

- ◆ 提取候选框的特征
- ◆ 建立观察模型
- ◆ 然后对这些候选框评分，并不断更新模型



## 动态跟踪流程—步骤3

- ◆ 找一个得分最高的候选框作为预测的目标 (Prediction A)
- ◆ 产生多个预测结果
- ◆ 对多个预测值进行融合得到更优的预测目标。

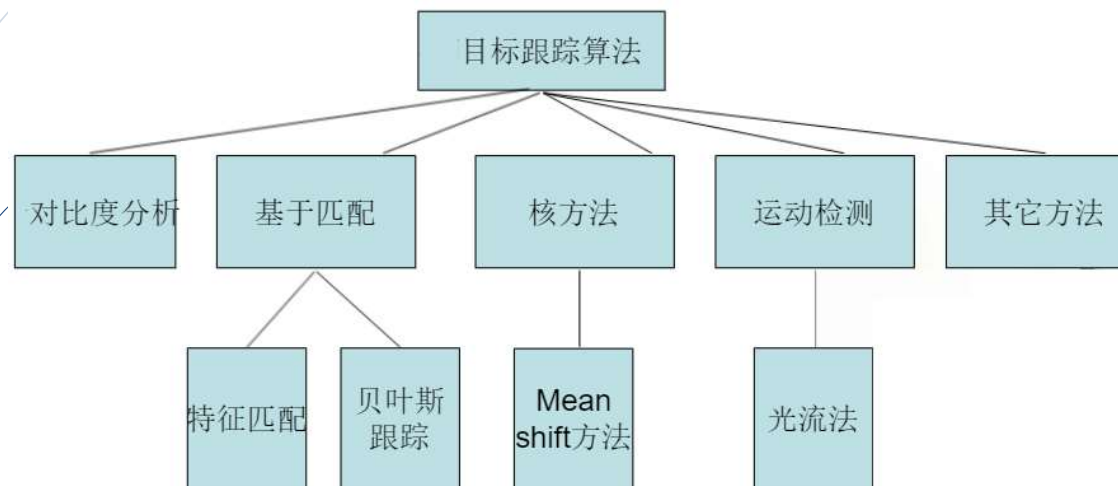


## 6. 视觉跟踪算法

- ◆ 生成式跟踪方法：采用稀疏编码，前些年曾经是研究的热点
- ◆ 判别式跟踪方法：近来逐渐占据主流地位，以相关滤波（Correlation Filter）和深度学习为代表，得了令人满意的效果。

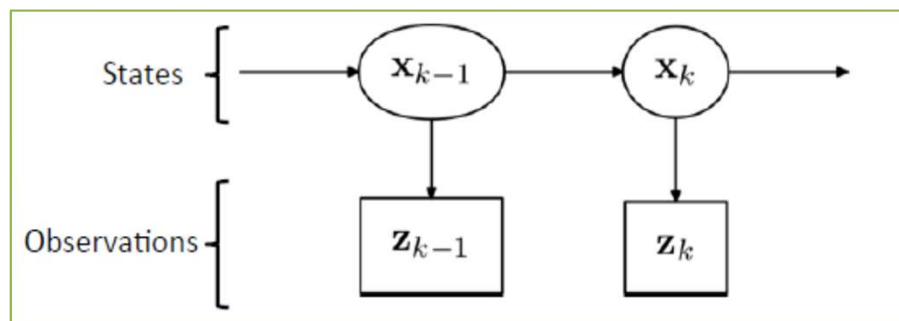


## 典型的视觉跟踪算法



## 贝叶斯跟踪算法框架

单目标跟踪算法，似然跟踪理论，根据 $k-1$ 时刻的状态推断 $k$ 时刻的状态：



## 典型的视觉跟踪算法

- ◆ 均值漂移算法 (Mean-shift)
  - 非参数化的特征空间
  - 求密度的局部极大
  - 颜色直方图
- ◆ 卡尔曼滤波器 (Kalman filter)
- ◆ 基于核的跟踪方法 (Kernel-based tracking)





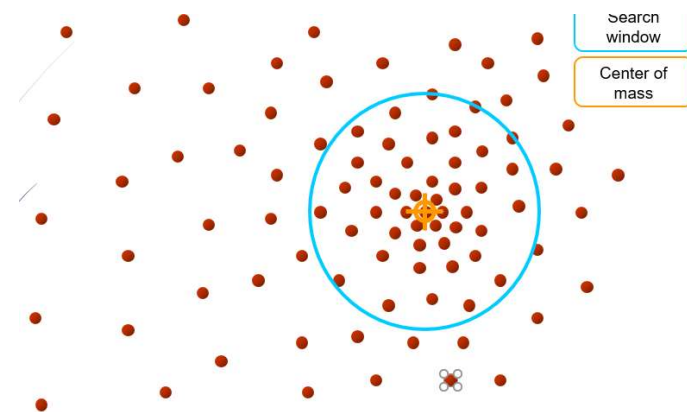
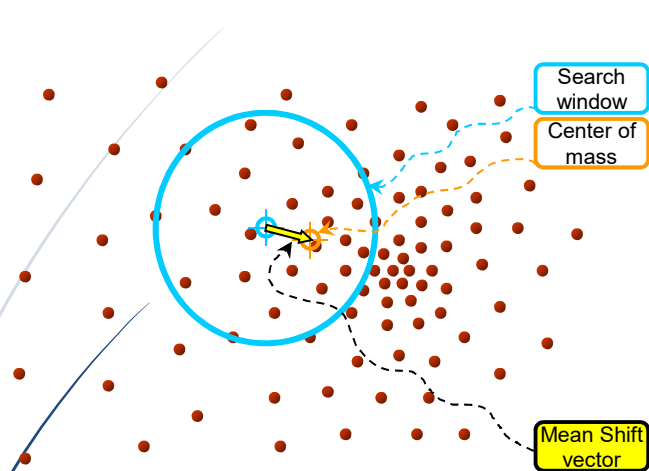
## (1) 均值漂移算法

- ◆ 均值移动的理论基础是概率密度估计：在概率密度空间中寻找局部极大点，算法步骤：
  - (a) 计算以初始点 $x_0$ 为中心的某一核窗所对应的均值移动向量 $m(x_0)$ ；
  - (b) 根据 $m(x_0)$ 来移动核窗的中心位置，也即把 $m(x_0)$ 中的加权平均值部分赋予 $x_0$ ，把 $x_0$ 作为新的初始点，并转回步骤(a)
  - (c) 重复(a)、(b)过程，直到满足某一预定的条件。

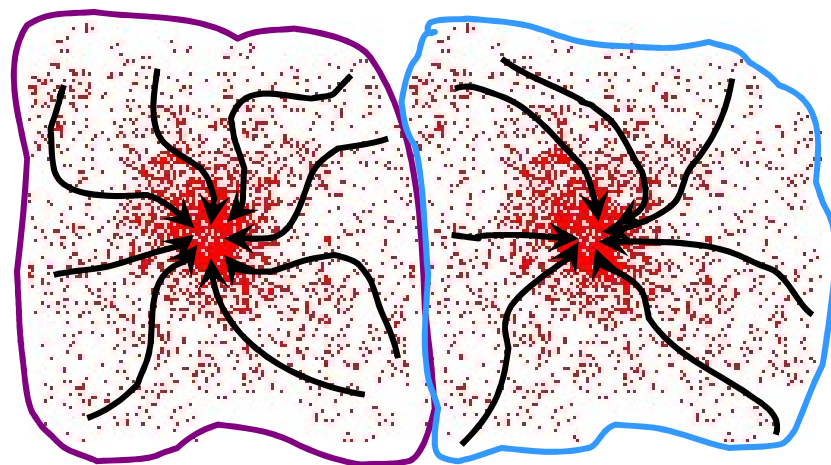
均值移动过程就是寻找数据分布最密处的过程。



## 均值漂移



## 均值漂移



- ◆ 漂移过程建立函数：对应跟踪对象位置的像素，值为1；其它位置为0
- ◆ 根据跟踪的像素与原目标像素计算相似性概率，需要构造的函数值与计算的概率值成正比。
- ◆ 似然值可以根据以下属性进行计算：
  - 颜色color
  - 纹理texture
  - 边缘形状 shape (boundary)
  - 预测位置 predicted location



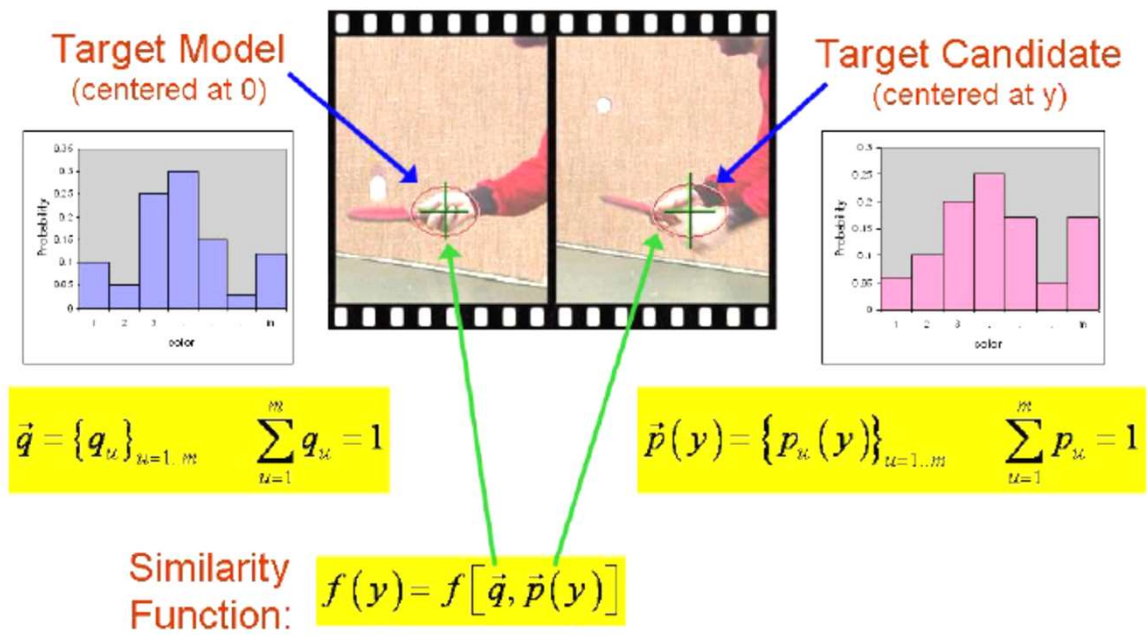
## 基于颜色的均值漂移模型

### ◆ 两种方法处理颜色：

- 创建颜色“似然”图像，像素通过与所需颜色的相似性加权
- 用直方图表示颜色分布。使用mean shift查找颜色分布最相似的区域。



# 均值漂移算法



## 相似性函数的计算方法

### Bhattacharyya Distance:

Given an n-bucket model histogram  $\{m_i \mid i=1, \dots, n\}$  and data histogram  $\{d_i \mid i=1, \dots, n\}$ , we follow Comanesciu, Ramesh and Meer \* to use the distance function:

$$\Delta(m, d) = \sqrt{1 - \sum_{i=1}^n \sqrt{m_i \times d_i}} \quad \left. \vphantom{\Delta(m, d)} \right\} \begin{array}{l} \text{Similarity Function} \\ f(y) = f[\vec{p}(y), \vec{q}] \end{array}$$

Dorin Comanesciu, V. Ramesh and Peter Meer, "Real-time Tracking of Non-Rigid Objects using Mean Shift," IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina, 2000

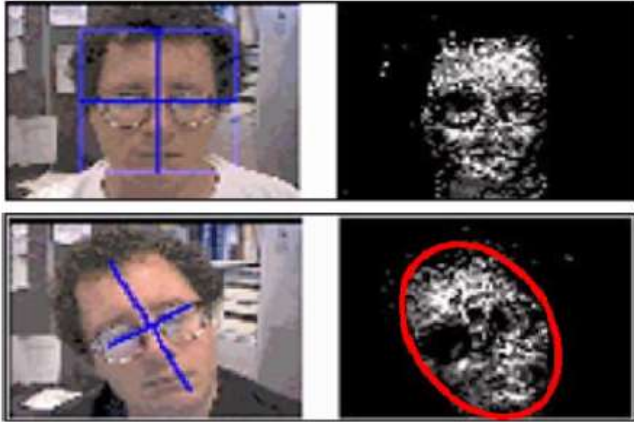


## 实例：Bradski's CamShift

- ◆ CamShift: Continuously Adaptive Mean-SHIFT
- ◆ 基本思想: 视频图像的所有帧作MeanShift运算，并将上一帧的结果（即Search Window的中心和大小）作为下一帧 MeanShift算法的Search Window的初始值，如此迭代。







Fast motion



Distractors



## 均值漂移算法优缺点

### ◆ 优点:

- 跟踪速度快
- 不需要纹理
- 允许目标外观略有变化

### ◆ 缺点:

- 单一的直方图不能捕捉外观的变化
- 有限的背景分辨力



## (2) Kalman滤波器的跟踪

### ◆ 卡尔曼滤波 (Kalman filtering)

- 斯坦利·施密特 (Stanley Schmidt) 首次实现了卡尔曼滤波器。
- 一种利用线性系统状态方程，通过观测系统输入输出数据，对系统状态进行最优估计的算法。



## (2) Kalman滤波器的跟踪

◆ 理论上，kalman滤波器需要三个重要假设：

- 1) 被建模的系统是线性的
- 2) 影响测量的噪声属于白噪声
- 3) 噪声本质上是高斯分布的



## (2) Kalman滤波器的跟踪

◆ 在kalman滤波器应用时，一般考虑三种运动：

- 1) 动态运动：这种运动是前次测量时系统状态的直接结果
- 2) 控制运动：这种运动是由于某种已知的外部因素以某种原因施加于系统，控制会改变系统运动
- 3) 随机运动：需要在预测阶段加入的随机运动。



## Kalman滤波器的跟踪

◆ Kalman滤波器的动态模型框架，Kalman滤波器的 $k$ 时刻的真实状态时从 $k-1$ 时刻演化而来，包括：

- 状态转移模型(状态转移矩阵)
- 控制向量，即施加的外部作用
- 过程噪声



## Kalman滤波器的跟踪

◆ 预估器:分为两个阶段:

- 第一阶段:预测阶段, 滤波器使用上一状态的估计, 作出对当前状态的估计。
- 第二阶段:更新阶段, 利用当前观测值优化在预测阶段获得的预测值, 以获得一个更精确的新估计值。



## Kalman滤波器的跟踪

两个重要方程：

状态方程：
$$x_k = Ax_{k-1} + Bu_{k-1} + q_{k-1}$$

观测方程：
$$y_k = Hx_k + r_k$$

$x(k)$ 是k时刻的系统状态  $u(k)$ 是k时刻对系统的控制量

A是传输参数、B是控制参数 对于多模型系统，他们为 矩阵

$y(k)$ 是k时刻的测量值，H是测量系统的参数，对于多测量系统，H为状态转移矩阵。

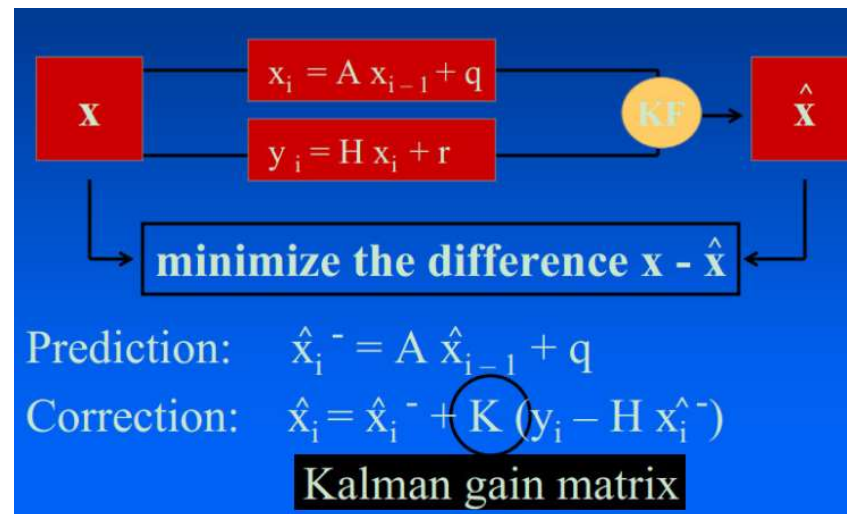
$q(k)$ 和 $r(k)$ 分别表示状态和测量的噪声。他们被假设 成高斯白噪声，他们的协方差分别是Q，R

假设他们不随系统状态变化而变化





## Kalman滤波器的跟踪



### (3) 在线跟踪算法

- ◆ 判别模型 (tracking-by-detection), 在线判别学习
- ◆ 在线检测、学习, 并预测
- ◆ 在线更新分类器

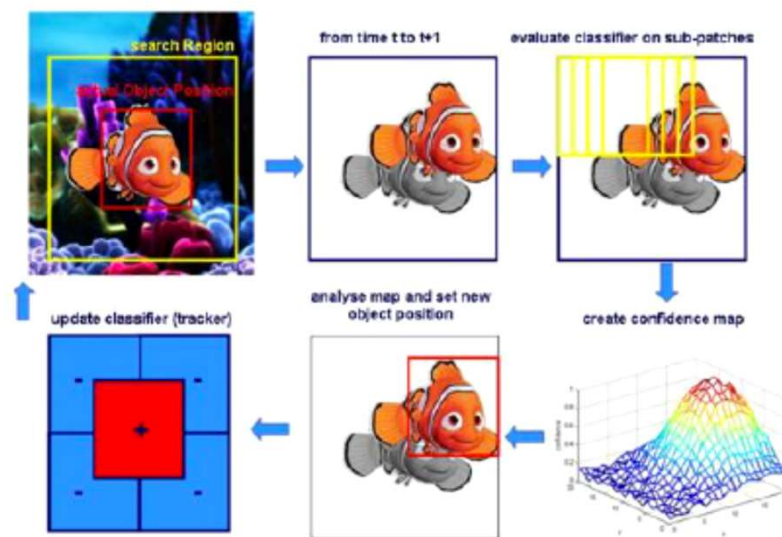
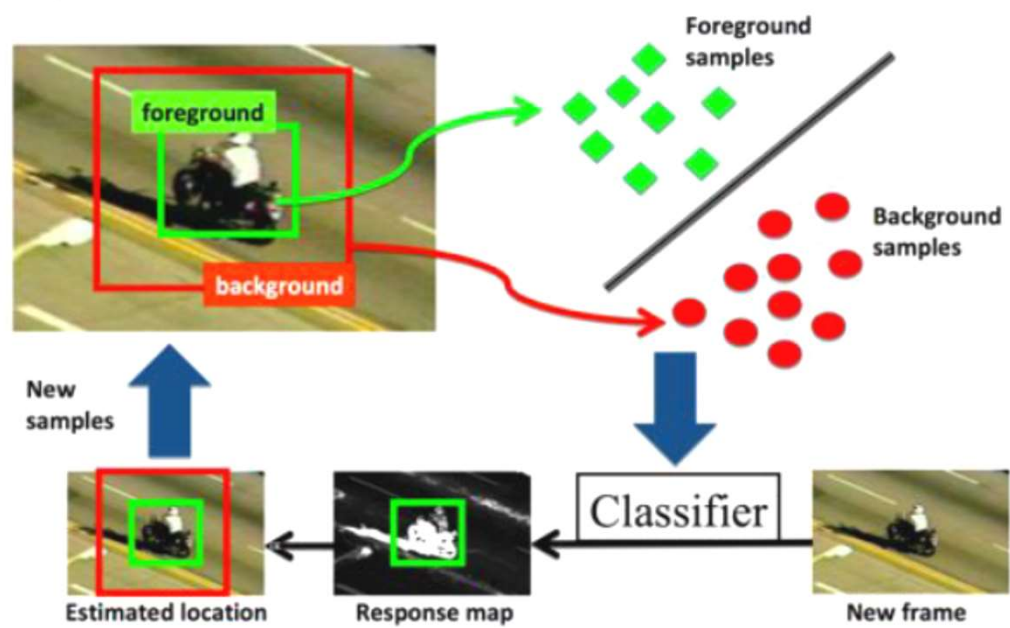


Figure from Grabner and Bischof CVPR 06



### (3) 在线跟踪算法



Slide from Collins, PSU



## 相关滤波器跟踪算法

- ◆ 相关滤波用于表示两个信号之间的相似程度，通常用卷积表示相关操作。
- ◆ 相关滤波的基本思想就是：寻找一个滤波模板，让下一帧的图像与滤波模板做卷积操作，响应最大的区域则是预测的目标。



## 相关滤波器跟踪算法

### ◆ CFT主要优点:

- 引入更好的训练结果
- 提取更强大的特征
- 结合部分跟踪策略，即相对于对目标整体识别，可以将目标分成几个部分，对各个部分进行识别



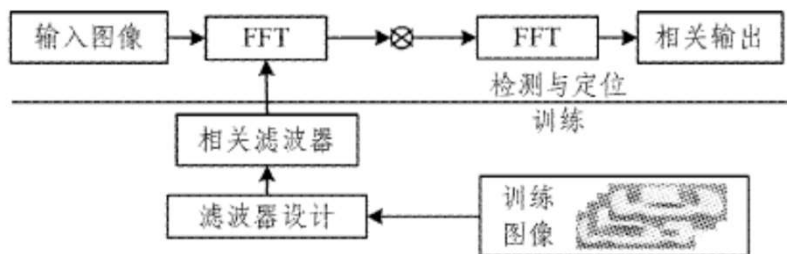
## 相关滤波器算法的思想

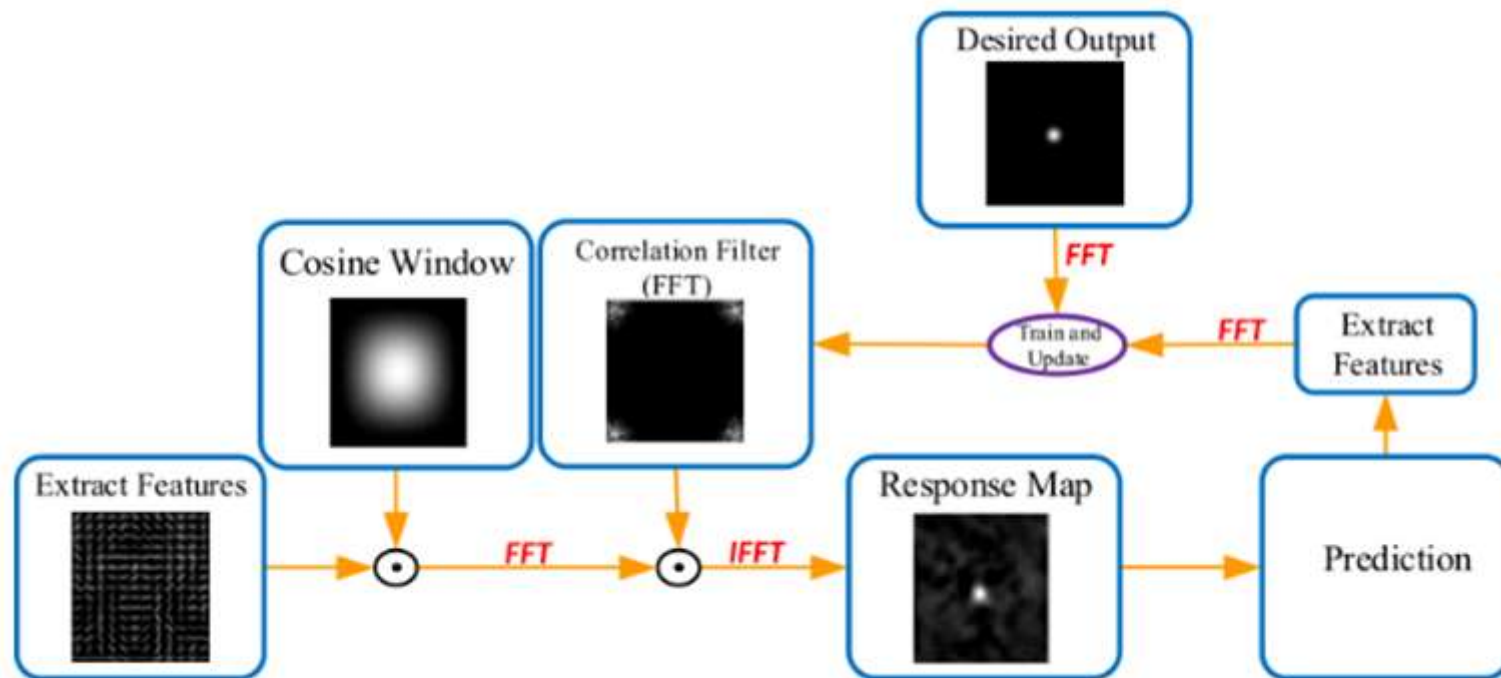
- ◆ 总体思路：根据当前帧的信息和之前帧的信息训练出一个滤波器，然后与新输入的帧进行相关性计算，得到的置信图就是预测的跟踪结果
- ◆ 显然，得分最高的那个点（或者块）就是最可能的跟踪结果



## 相关滤波器算法的思想

- ◆ 对于输入的第一帧，将给定的要追踪的区域提取出特征，然后进行训练，得到相关滤波器
- ◆ 相关滤波器算法的思想：学习一个滤波器 $h$ ，然后和图像 $f_i$ 进行卷积操作
$$g_i = f_i * h$$
- ◆ 最大响应点所在的区域即为要追踪目标的新位置，然后再用新位置区域训练更新得到新的相关滤波器，用于之后的预测。







## (4) 多目标跟踪

描述:

- ◆ 输入: 一系列检测结果 (from next module R-CNN)
- ◆ 输出: 对应每个检测的状态 (id)



## 多目标跟踪

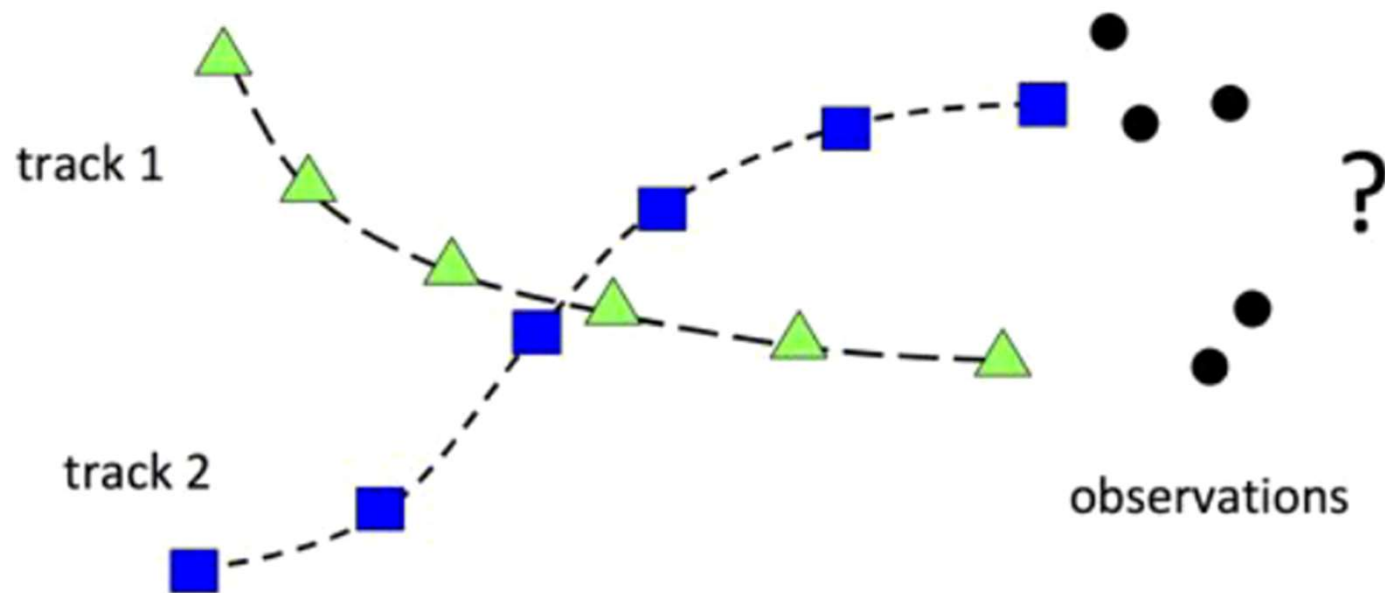
- ◆ 主要任务是给定一个图像序列，找到图像序列中运动的物体，并将不同帧中的运动物体一一对应(**Identity**),然后给出不同物体的运动轨迹
- ◆ 如行人、车辆、运动员等



## 多目标跟踪问题

- ◆ 数据关联
- ◆ 指派问题
- ◆ 离散组合优化

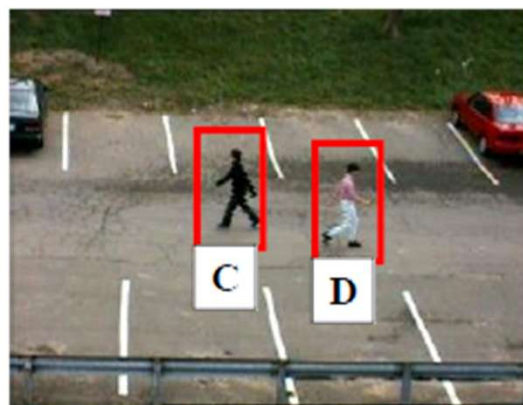
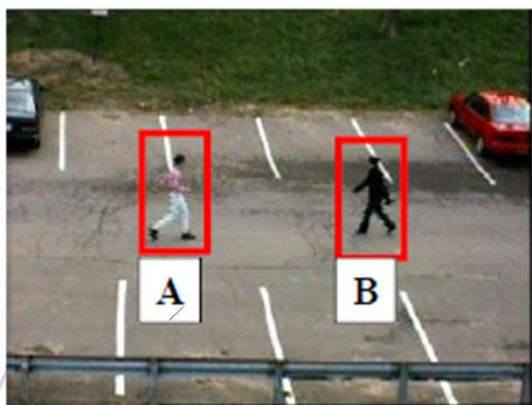




Slide from Collins, PSU



## 目标合并或分离后的数据关联实例



$$\Delta(A,C) = 2.03$$

$$\Delta(A,D) = 0.39 \quad \bullet$$

$$\Delta(B,C) = 0.23 \quad \bullet$$

$$\Delta(B,D) = 2.0$$

A  $\rightarrow$  D

B  $\rightarrow$  C

$$\Delta(m,d) = \sqrt{1 - \sum_{i=1}^n \sqrt{m_i \times d_i}}$$



## 7. 深度学习的跟踪方法

- ◆ 将分类图像数据集上训练的卷积神经网络迁移到目标跟踪问题中，基于深度学习的目标跟踪方法得到快速发展
  - CNN-SVM利用在ImageNet分类数据集上训练的卷积神经网络提取目标的特征，再利用传统的SVM方法做跟踪。
  - FCN用目标的两个卷积层特征构造了可选特征图的网络，进一步提升性能
  - 充分利用卷积神经网络各层特征，在相关滤波的基础上结合多层次卷积特征，提升跟踪效果



## 深度学习跟踪方法

- ◆ 分类任务注重类间差异，忽视类内的区别。目标跟踪任务关心的则是区分特定目标与背景，抑制同类目标。因此，分类数据集上预训练的网络可能并不完全适用于目标跟踪任务。
- ◆ Nam设计了一个专门在跟踪视频序列上训练的多域（Multi-Domain）卷积神经网络（MDNet），取得了VOT2015比赛的第一名
- ◆ 2016年SRDCF的作者利用了卷积神经网络提取目标特征，然后结合相关滤波提出了C-COT的跟踪方法，取得了VOT2016的冠军



## 目标跟踪的现有工作

### ◆ 单目标跟踪综述文献:

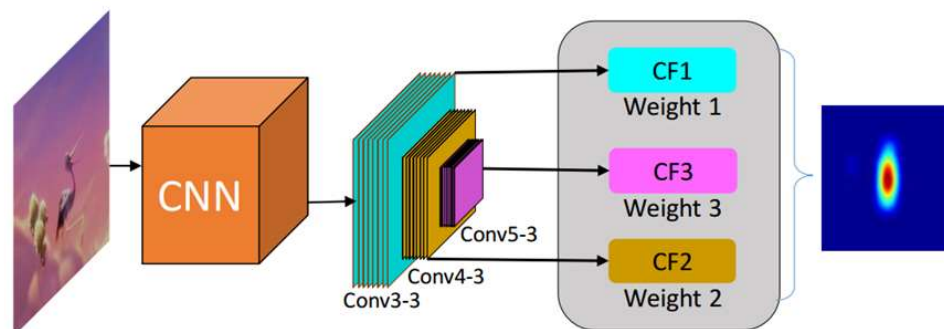
《Handcrafted and Deep Trackers: A Review of Recent Object Tracking Approaches》





## 单目标跟踪—HCF

- ◆ C. Ma, J. B. Huang, X. Yang, and M. Yang, “Hierarchical convolutional features for visual tracking,” in ICCV. IEEE, 2015, pp. 3074 – 3082.
- ◆ 利用VGGNet-19提取深度特征，并选取conv3-4、conv4-4和conv5-4三层的特征(大小为[37, 28, 19])，并通过双线性插值将三层特征图调整至同一大小，然后分别对每层特征图做相关滤波操作，最后加权融合(权重为[1, 0.5, 0.02])三层特征对应的置信度图，得到最终的置信度图。



## 单目标跟踪—HCFT\*

- ◆ 增加了尺度估计和重捕功能

