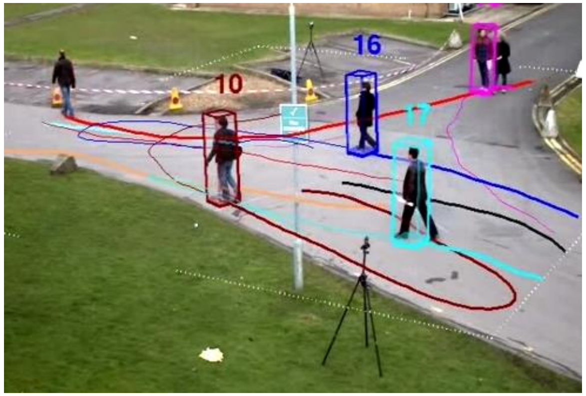
Class 9

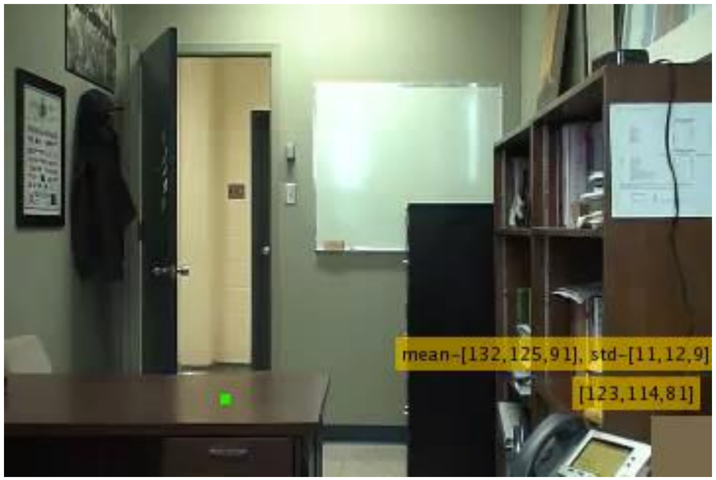
Change Detection

Tracking

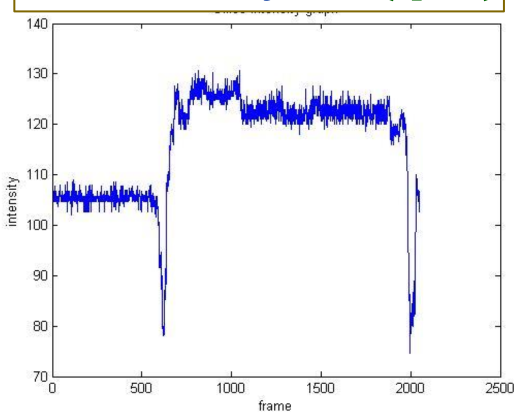


Parametric Pixel Modeling

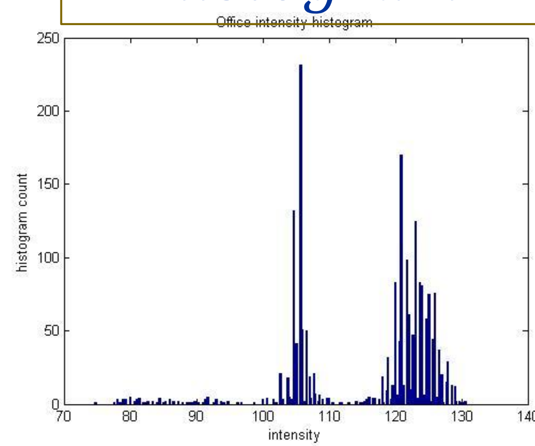
Background Pixel



Intensity: $I(q,t)$

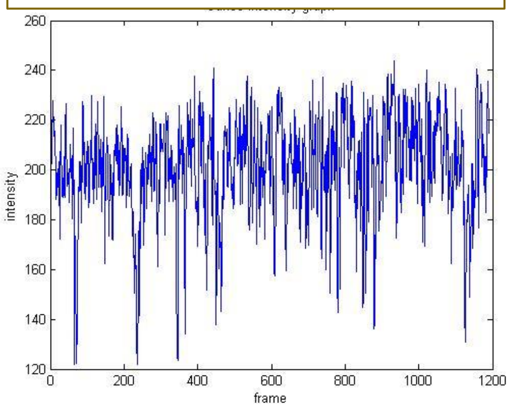


histogram

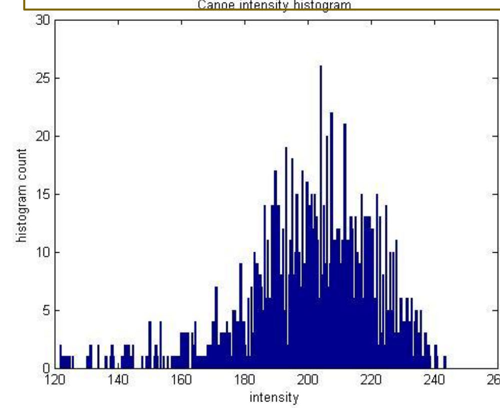


Example 2

$Intensity$: $I(q,t)$



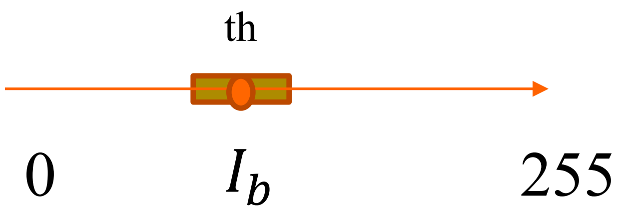
$Histogram$

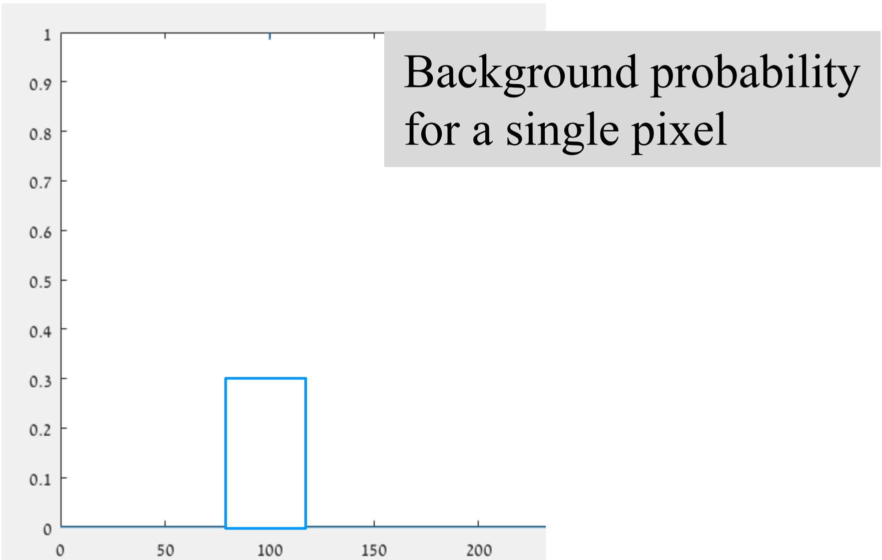


Background Distribution

- Let $P_B(x) = P(x|x \in B)$ be the probability distribution function (*pdf*) of the background for a pixel q
- Given a threshold α, and $\mathrm{x_t} = I(q,t)$:

$$F(x_t) = \begin{cases} 1 & P_b(x_t) < \alpha \\ 0 & P_b(x_t) \geq \alpha \end{cases}$$





Background Distribution

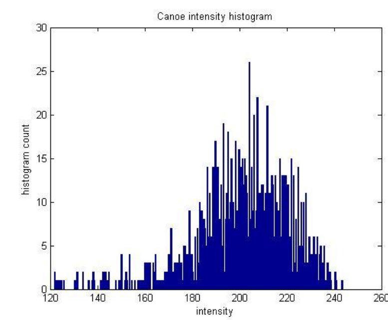
- Let q be a pixel, and $\mathrm{x_t} = I(q,t)$ be the intensity of q at time t
- Define the probability distribution function (*pdf*) of the background, using $\{I(q,t)\}$:

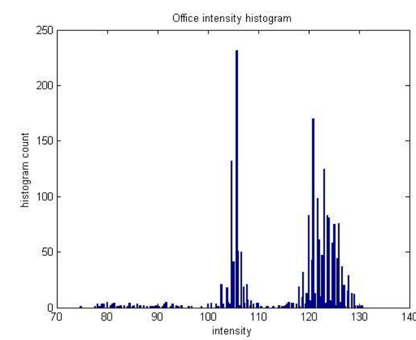
$$P_B(x) = P(x|x \in B)$$

Model the *pdf*

- Let $x_t = I(q,t)$
- Regard the histogram of $\{x_t\}_{t=0}^{n}$ as a *pdf* of q background
- Parametric models:
  - 1D/3D Gaussian
  - Multi modal Gaussians
  - Others…
- Non-parametric models

Under which assumption is it true?



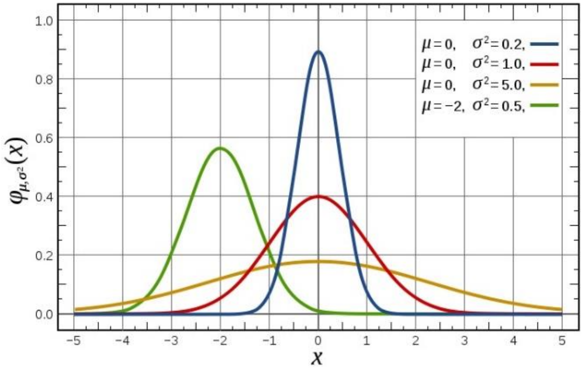


Model: 1D Gaussian

- Assume: independent Gaussian noise in the sampling process:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

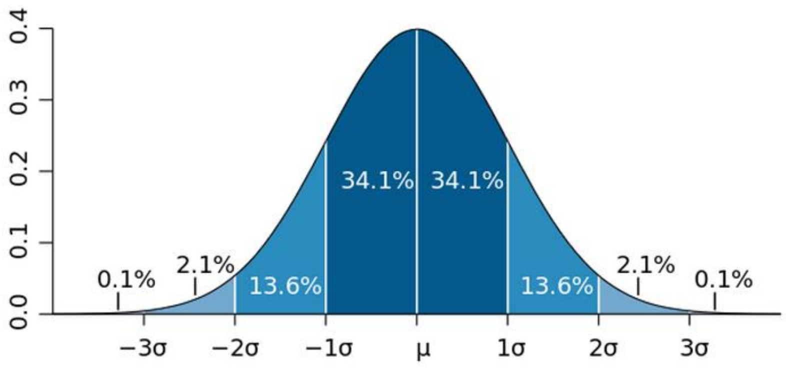
- Parameters:
  - $\mu = E(x)$ - mean (expectation)
  - σ - STD
  - $\sigma^2(x) = E[(x-\mu)^2]$ - Variance



How to set the threshold α?

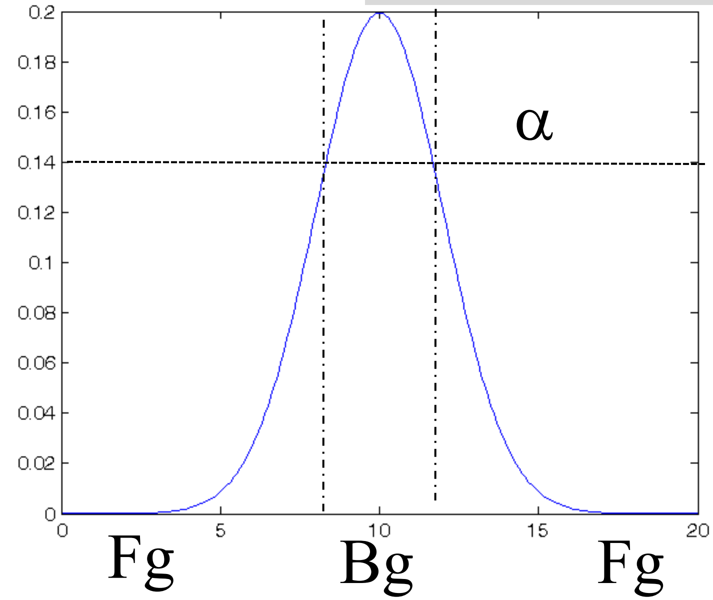
- $x_t = I(q,t)$
- $F(x_t) = \begin{cases} 1 & P_b(x_t) < \alpha \\ 0 & P_b(x_t) \geq \alpha \end{cases}$
- A common choice:

  $\alpha = 2.5\sigma_{i,t}$



Where are the failures?

Background probability for a single pixel



Gaussian Mixture Model
(based on Stauffer et. al. 1999)

- Motivation:
  - Moving background – e.g., trees
  - Single Gaussian is insufficient
- Use mixture of Gaussian:

$$P(x_t \mid B) = \sum_{i=1}^{K} w_{i,t} G(x_t, \mu_{i,t}, \sigma_{i,t})$$

K Depends on memory and computational power

  - K: number of Gaussians
  - $w_{i,t}$: weight of the i^{th} Gaussian at time t
  - $\mu_{i,t}$ *and* $\sigma_{i,t}$: the i Gaussian parameters

Issues

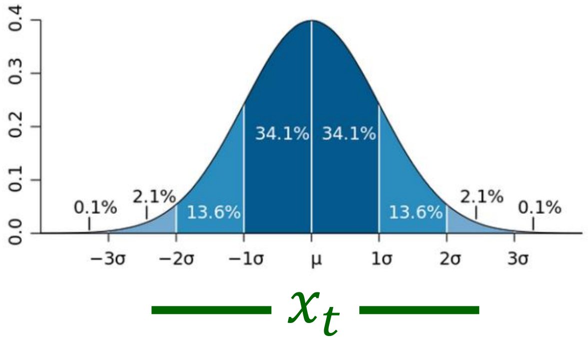
- How to use the set of K Gaussians
- How to initialize the Gaussian parameters:

  For a single Gaussian, i :
  - Average: $\mu_{i,t} = average(\{x_t\})$
  - Variance: $\sigma^2 = E[(x_t - \mu)^2]$
- How to update the Gaussian parameters
- Note: we first assume grey-level images

Match x_t with G_i

- Define, x_t match G_i by:

$$M(x_t, Gi) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$



- Assume G_i is a background model of q, and $x_t = I(q, t)$ then we consider x_t to be a background pixel if $M(x_t, G_i) = 1$

Using the set of K Gaussians

- Given the set B of dominant Gaussians:
  - x_t is foreground: $M(x_t, G_i) = 0\ , \forall G_i \in B$
  - x_t is background: otherwise

Weight of G_i

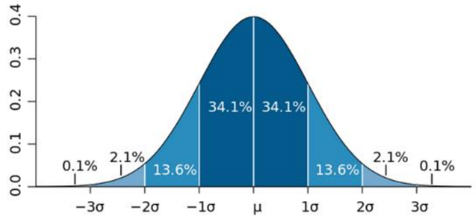
- Let w_{it} be the weight of G_i with σ_{it}
- Order the set of K Gaussians by :
  w_{it}/σ_{it}
  - high: more evidence & low variance
- Use it to define the set of dominant Gaussians

  (details in the paper)

Update μ_i & σ_i

$$\sigma^2 = E[(x_t - \mu)^2]$$

- Matched: $M(x_t, G_i) = 1$
  - $\mu_{i,t} = (1-\rho)\mu_{i,t-1} + \rho x_t$
  - $\sigma_{i,t}^2 = (1-\rho)\sigma_{i,t-1}^2 + \rho\left(x_t - \mu_{i,t}\right)^2$
  - ρ is the learning rate defined by a parameter α : $\rho = \alpha G(x_t \mid \mu_i, \sigma_i)$
- Unmatched: remains the same



$$M(x_t, G_i) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$

Update Weights

- Update weights of all Gaussians:
  - $w_{i,t} = (1-\alpha)w_{j,t-1} + \alpha\left(M\left(x_t, G_{j,t-1}\right)\right)$
  - α is a learning rate parameter

- Renormalize the weights $\sum_j w_{j,t} = 1$

$$M(x_t, Gi) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$

When does $w_{j,t}$ increase ?

Update the Set G_i

- Given x_t such that $\forall i,\ \ M(x_t, G_i) = 0$
- Replaced G_i with smallest w_{it}/σ_{it} with a new Gaussian:
  - $\mu_{j,t} = x_t$
  - Set σ_j high

Why?

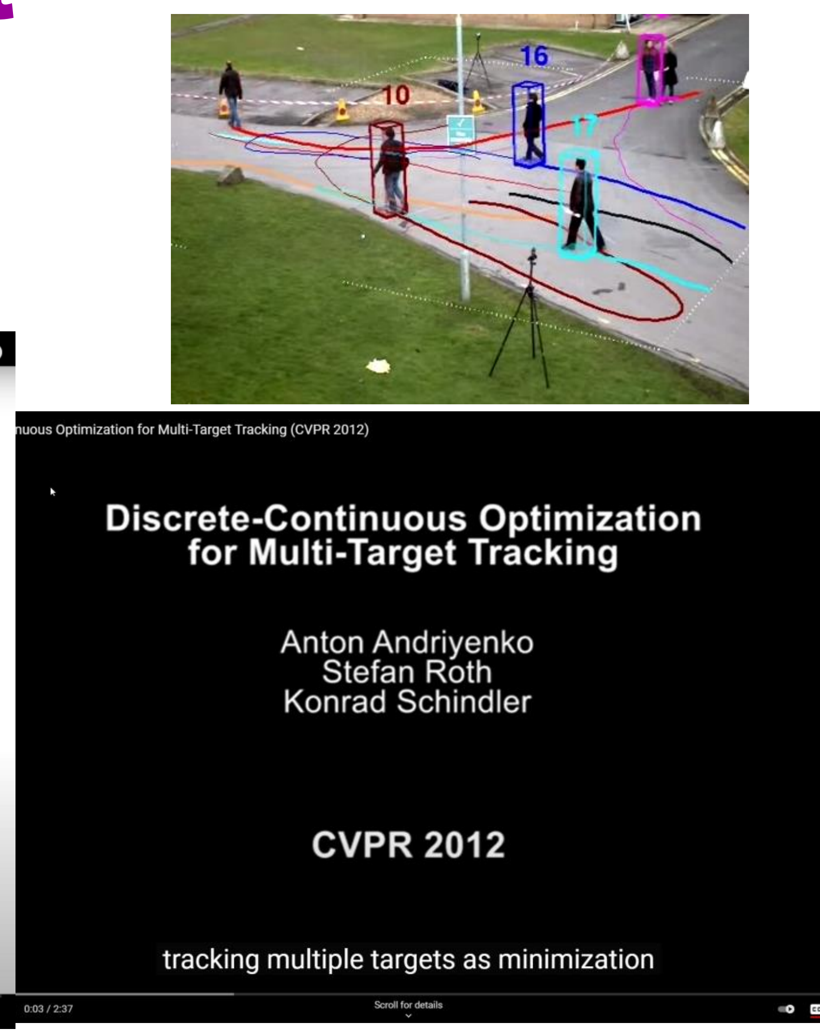
Summary

- Optical Flow:
  - Assumptions
  - Pairs of images
  - Multi scale
- Change Detection
  - Learn and model the background
  - Compare a frame to the BG
  - Mixture of Gaussians

Next

video

ealtime Multi-Person
Real-time Multi-Person 2D Pose Estimation Using Part Affinity Fields
Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh
Carnegie Mellon University
0:00 / 4:30
Scroll for details



16
10
nuous Optimization for Multi-Target Tracking (CVPR 2012)
Discrete-Continuous Optimization for Multi-Target Tracking
Anton Andriyenko
Stefan Roth
Konrad Schindler
CVPR 2012
tracking multiple targets as minimization
0:03 / 2:37
Scroll for details

Tracking

- Find location of a target in a sequence of images
- The egg and the chicken:
  - Perfect recognition implies tracking
  - Recognition often considered a more difficult problem

Tracking General Issues

The target

- What to track?
- How to detect it?
- How to represent it ?

Association

- Match detected target in different frames
- Build up a track history

Prediction

- Improve detection
- Improve efficiency

Tracking General Issues

Target: what?

- Feature points
- A region
- An object

Association

- Match detected target in different frames
- Build up a track history

Prediction

- Improve detection
- Improve efficiency

Tracking General Issues

Target: How to detect?

- Feature detection
- Object detection
- Using motion: e.g., optical flow or change detection
- Manually

Association

- Match detected target in different frames
- Build up a track history

Prediction

- Improve detection
- Improve efficiency

Tracking General Issues

Representation

- Feature points
- Patches
- A region
- Motion
- Learned e.g., discriminative from BG

Example of Target representation

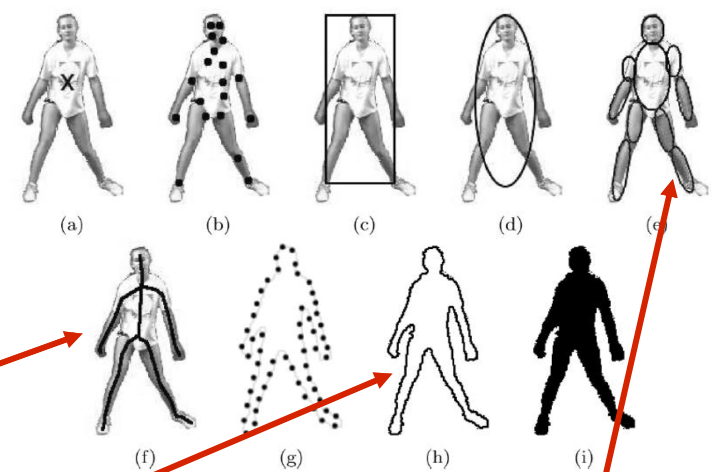


Fig. 1. Object representations. (a) Centroid, (b) multiple points, (c) rectangular patch, (d) elliptical patch, (e) part-based multiple patches, (f) object skeleton, (g) complete object contour, (h) control points on object contour, (i) object silhouette.

Skeleton

Outline: splines

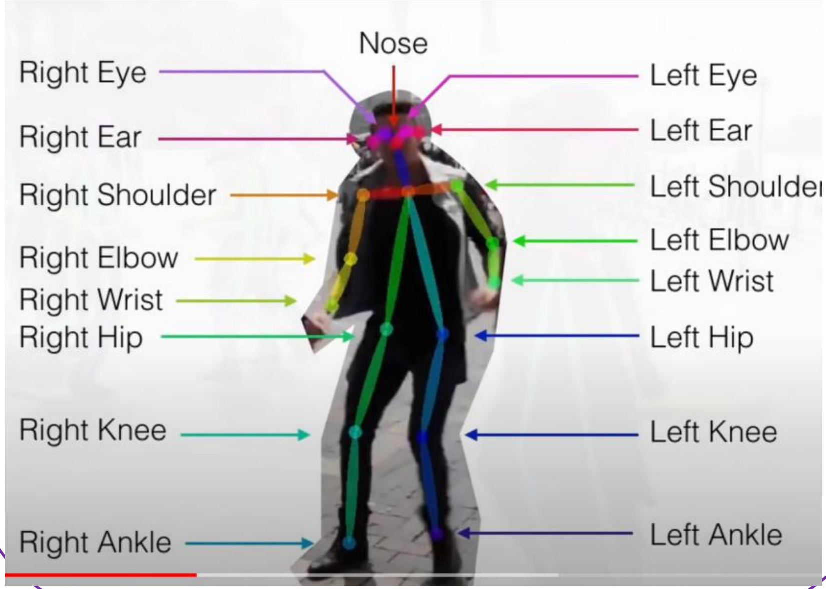
Object parts

Tracking General Issues

Representation

- Feature points
- Patches
- A region
- Motion
- Learned e.g., discriminative from BG

Example of Target representation



Tracking General Issues

Representation

- Feature points
- Patches
- A region
- Motion
- Learned e.g., discriminative from BG

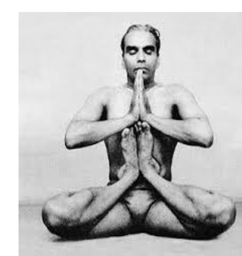
Representation Properties

- Unique
- Reliable location
- Easy to compute
- Insensitive to changes

Illumination

Non rigid

View point



Articulated

Tracking General Issues

The target

- What to track?
- How to detect it?
- How to represent it?

Association

- Low level: intensity, color, edges, patch descriptors (SIFTs, histograms, ..)
- High level: objects
- Learning methods
- Using motion (e.g., OF)
- By elimination

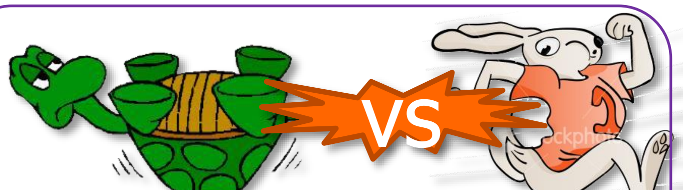
Association: Challenges

Occlusions

Many moving objects

Cluttered background

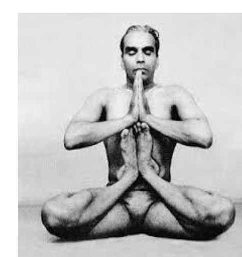
Similar objects



Time & Space complexity

Illumination

View point



Articulated

Tracking General Issues

The target

- What to track?
- How to detect it?
- How to represent it?

Association

- Match detected target in different frames
- Build up a track history

Prediction

- Improve detection
- Improve efficiency

Feature Based

- Use Lucas-Kanade OF to track corners (track with pure translation)
- Use affine registration with first feature patch
- Terminate tracks whose dissimilarity gets too large
- Start new tracks when needed

Tracking results

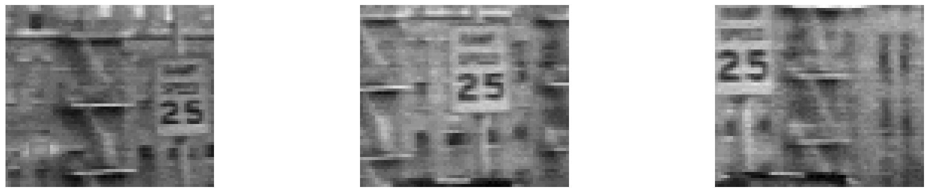


Figure 1: Three frame details from Woody Allen's *Manhattan*. The details are from the 1st, 11th, and 21st frames of a subsequence from the movie.

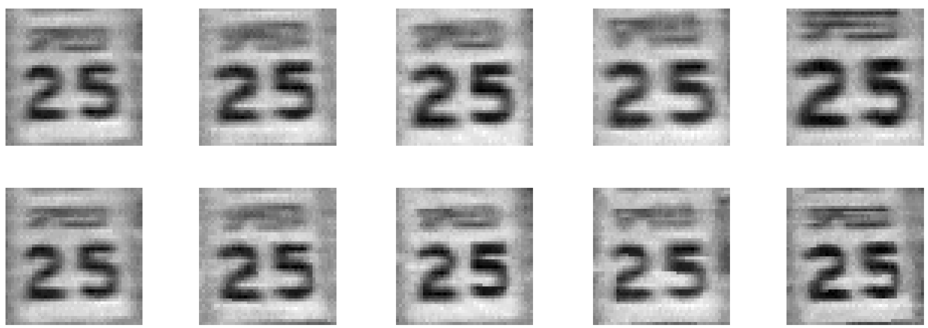
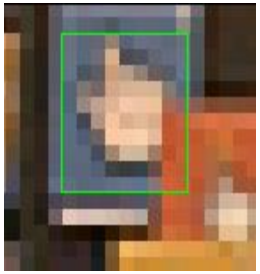
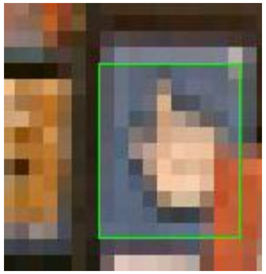


Figure 2: The traffic sign windows from frames 1,6,11,16,21 as tracked (top), and warped by the computed deformation matrices (bottom).

Region Based

- A template – the region to track
- Search for a match in the next image:
  - Use region descriptor
  - Use a distance (or similarity) measure
  - Limit the search area
- Choose the maximum (or minimum) as the match
- example





Window size

Small windows:

- More false positive matches
- Flow resolution – higher
- Cheap to compute

Large windows

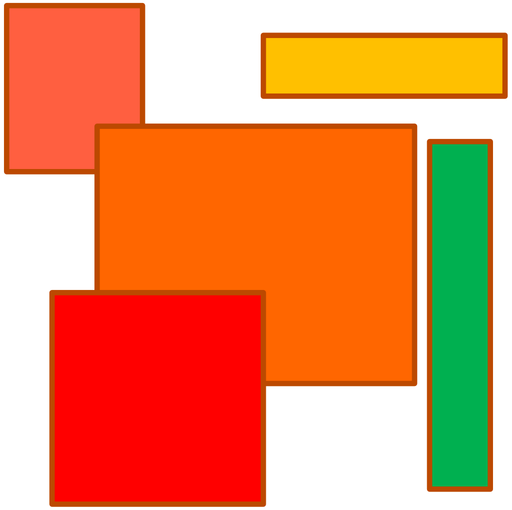
- More reliable
- Flow resolution – lower
- Expensive to compute

Improve Robustness

- Use "Good" patch descriptors
  - E.g., Histograms, Gradients, Histograms of gradients,
- Update the patch:
  - e.g., size: $s = \alpha s + (1 - \alpha)\, s\_t$
- Update the descriptor:
  - e.g., $\mathrm{d} = \alpha d + (1 - \alpha)\, d\_t$
- Use motion

Dealing with Occlusion
(based on Adam et al)

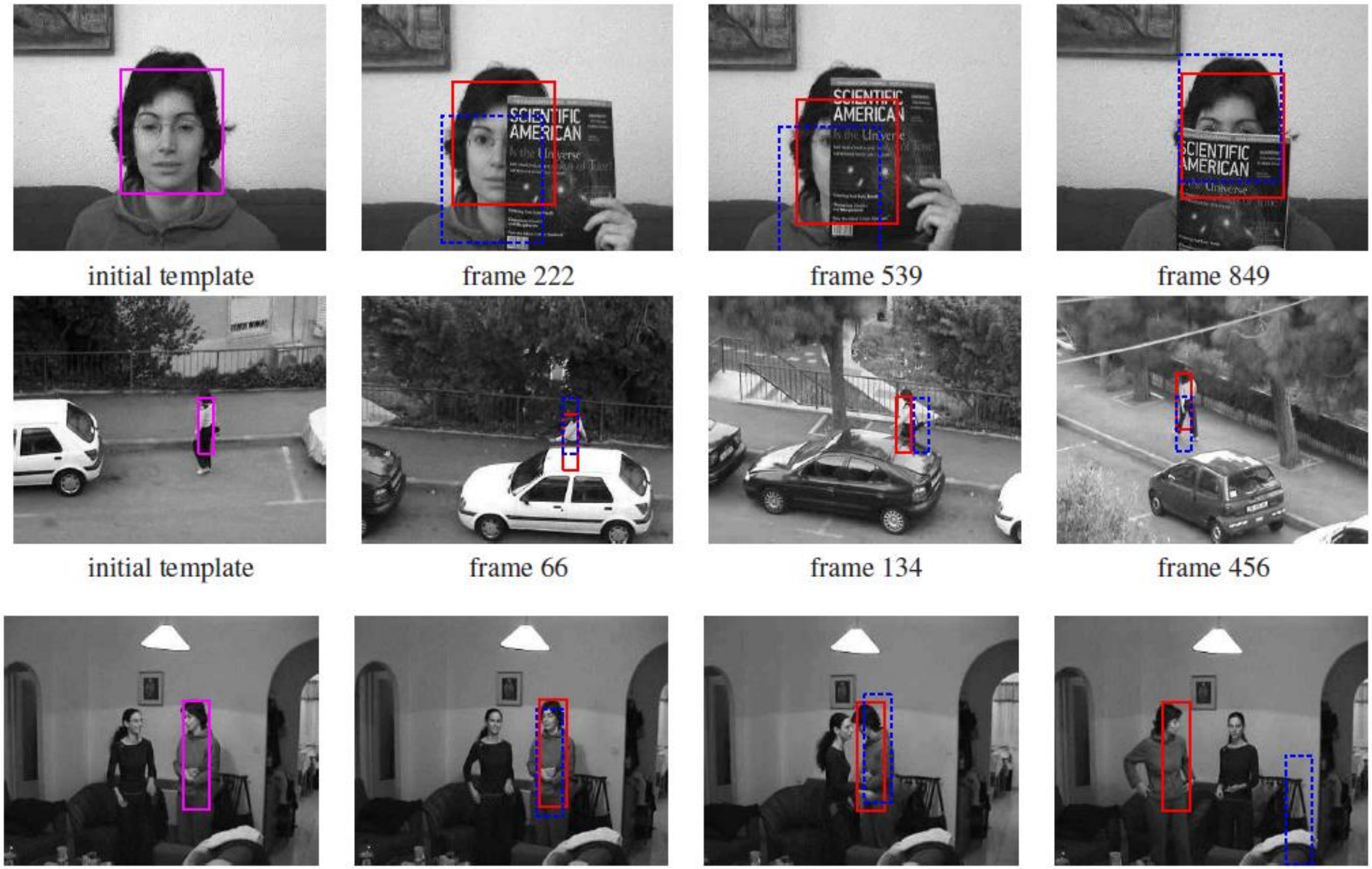
- Divide the region to several patches
- Calculate similarity measure for each patch
- Decide on the best object location by considering all the fragments response



Combining the Vote Maps

- Goal: deal with occlusion
- Consider the best Q patches
  - the maximal number of patches we always expect to be inliers
- Choose the location which maximizes the score of the best Q patches
- Vote for the location
  - How?

Results



initial template | frame 222 | frame 539 | frame 849

initial template | frame 66 | frame 134 | frame 456

initial template | frame 29 | frame 141 | frame 209

Results (cont)

initial template

frame 48

frame 82

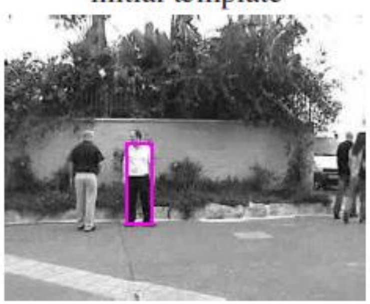
frame 110

initial template

frame 30

frame 100

frame 180



initial template

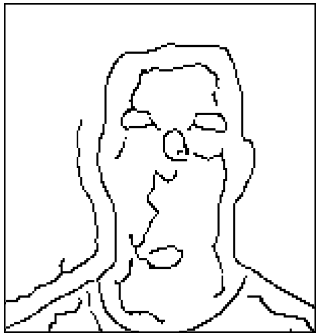
frame 35

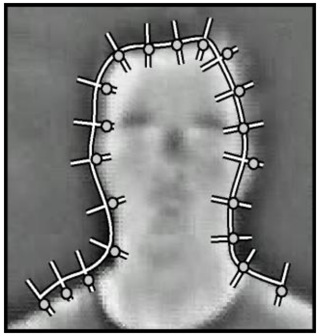
frame 65

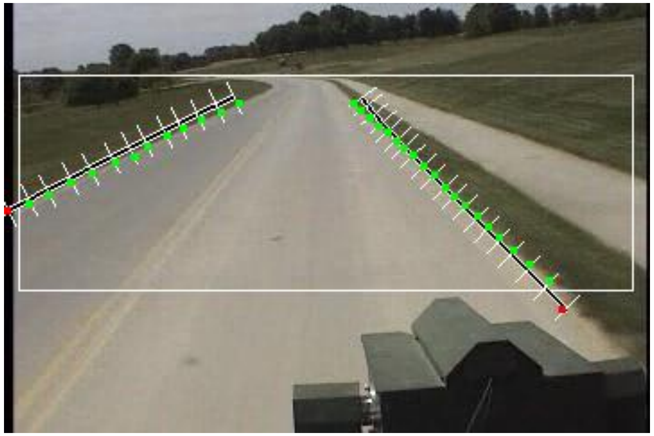
frame 90

Track Outlines

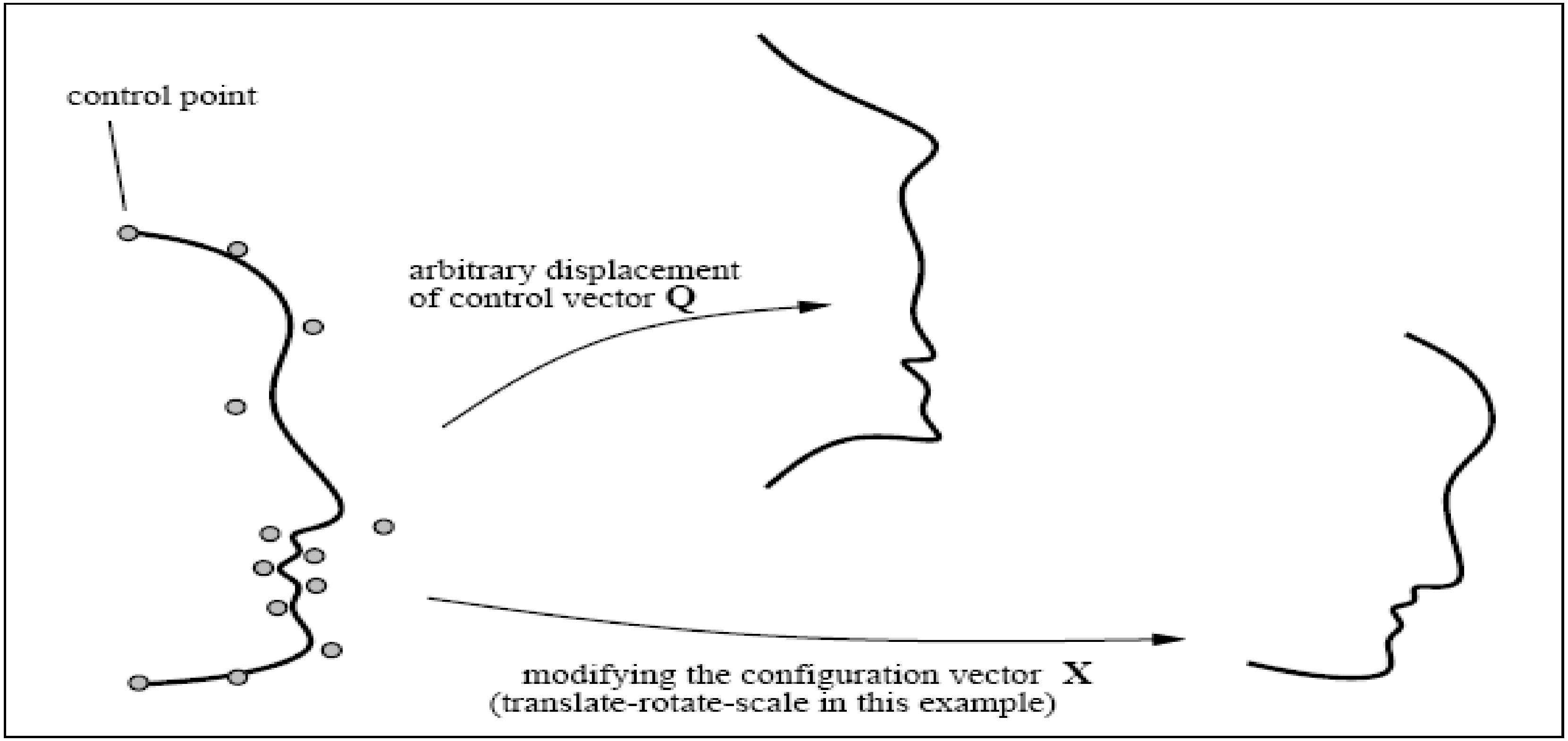
- Track contours using edge information:
  - Silhouettes
  - Road lines
- Outline representation:
  - E.g., by spline
- Find the outline:
  - E.g., Snakes



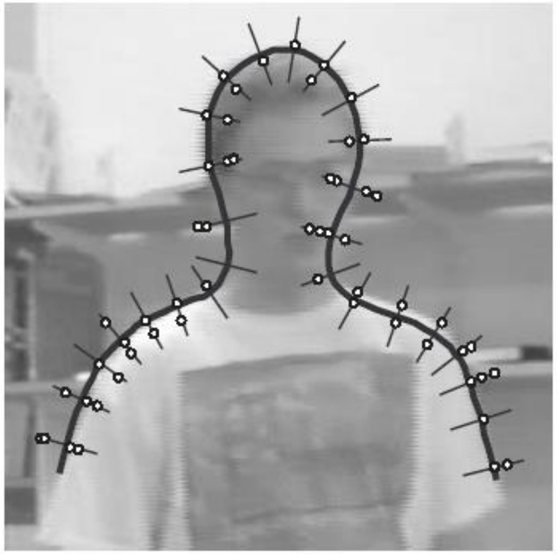




Shape Space



It is desirable to restrict the configuration of a spline



Constraint the Problem

- Knowledge about:
  - The object shape
  - The object appearance …
  - The motion: direction, velocity, …
- Helps:
  - Reduce ambiguity
  - Reduce the search space
- Obtain by:
  - Tailoring (hacking)
  - Learning

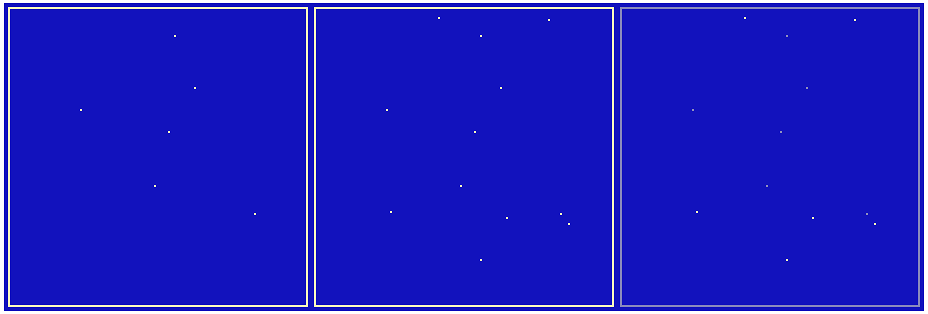
A ball, a person, a car

Color, texture, …

Used for prediction

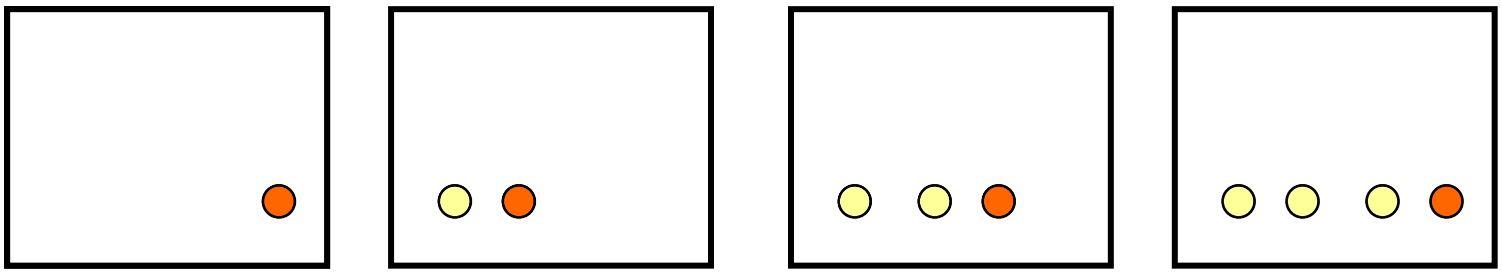
A Challenging Example

- The points are indistinguishable
- What assumptions can be used?
  - Smooth motion
  - Limited speeds
  - Short occlusions



Taken from http: http://visual.ipan.sztaki.hu/psmweb/

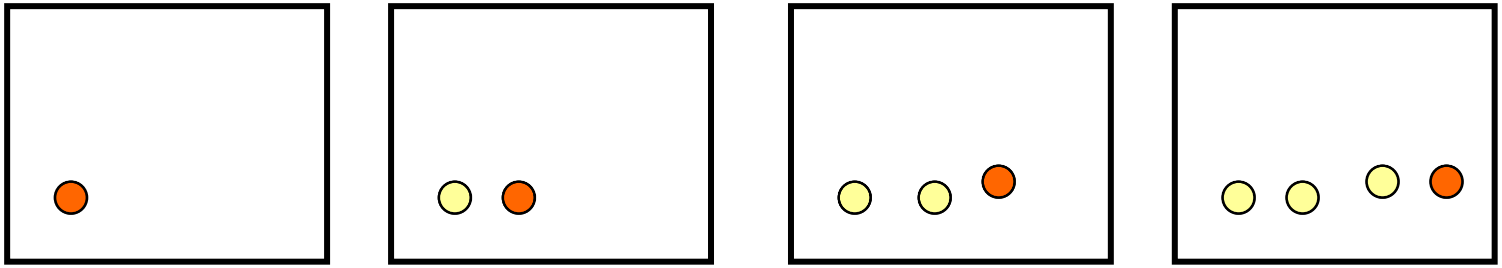
Simplify World



Assumptions:

- Linear 2D motion
- Constant velocity

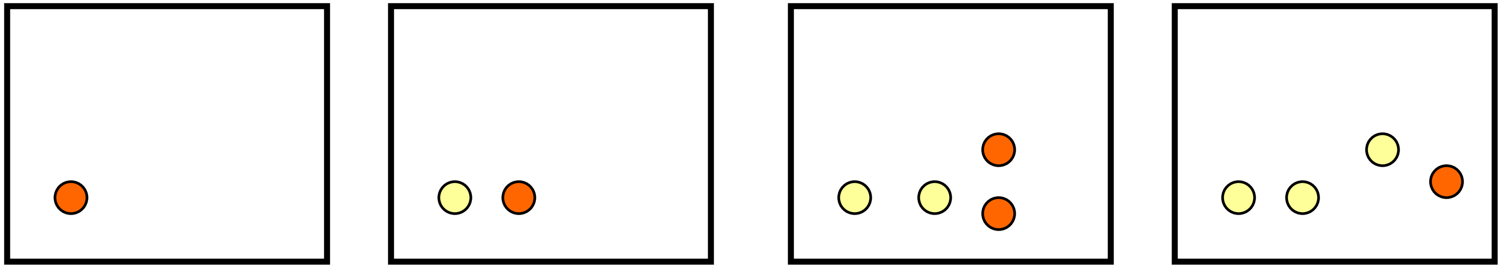
Noise



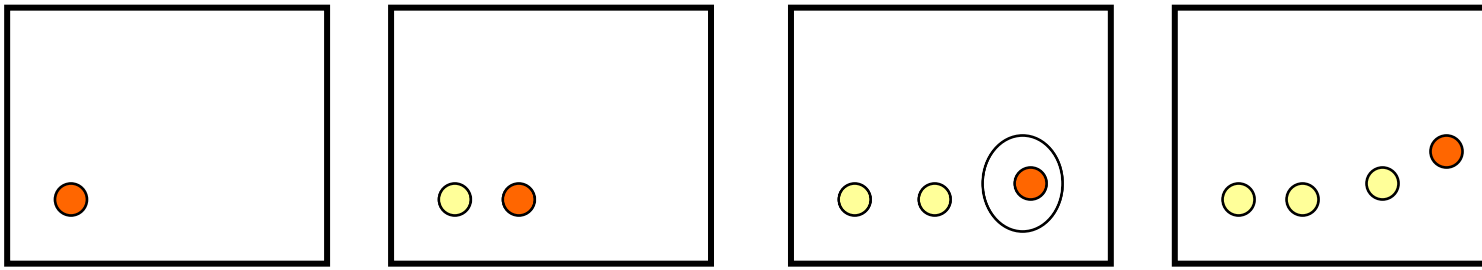
Assumptions:

- Linear 2D motion
- Constant velocity

Ambiguity



Use Prediction



- Based on motion:
  - Direction
  - Velocity
  - Acceleration
- Shape

Upadate predictions by measuments

Prediction

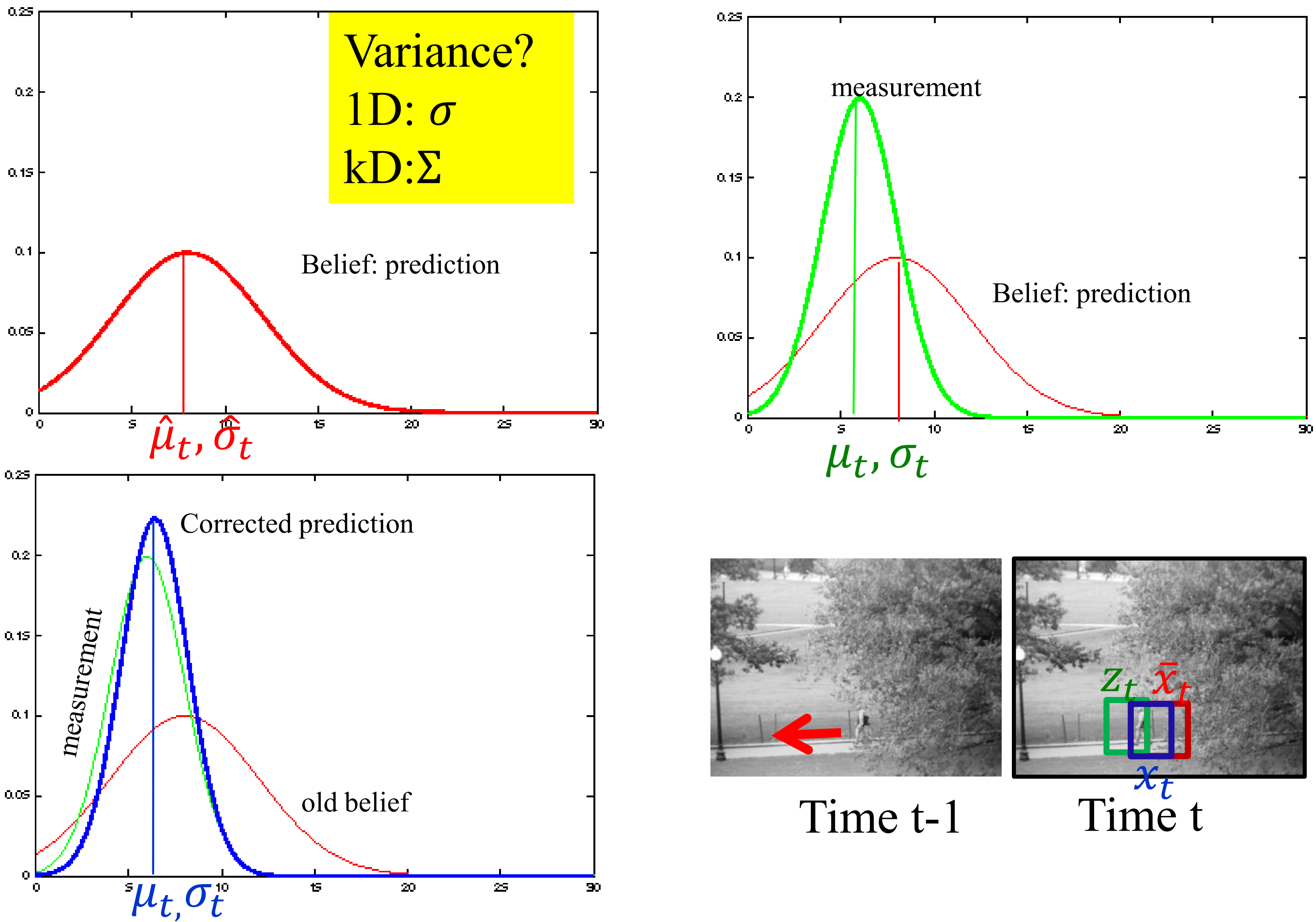
- Used for:
  - Improve accuracy
  - Reduce search space in tracking
- Based on what we saw and the transition model (e.g., motion)

Kalman & Particle Filters

- Basic idea:
  - Use noisy measurements from the image
  - Use prediction
  - Improve robustnedd by combining prediction and measurements

Time t-1 Time t

Kalman Filter

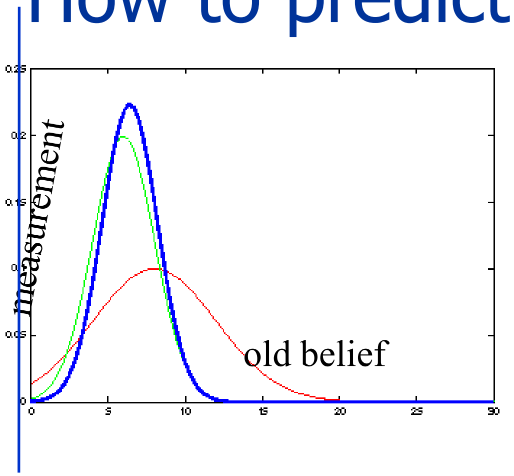


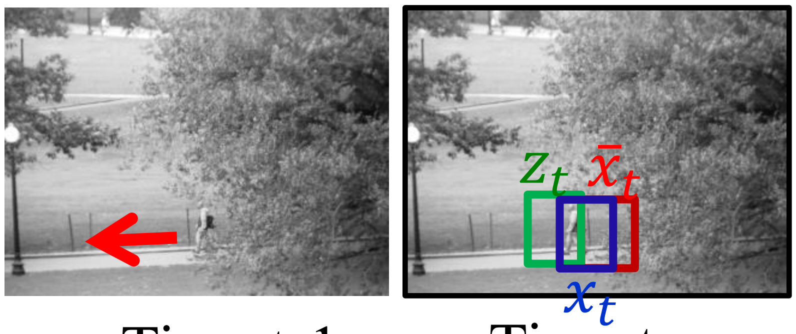
Kalman Filter

- Weighted sum: $\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-)$
- Questions:
  - How to set the weight between prediction and observations?
  - How to set the variance?
  - How to predict ?

$$k_G = \frac{E_{EST}}{E_{EST} + E_{MEA}}$$

$$E_{EST}(t) = (1 - k_G)E_{EST}(\text{t-1})$$





Time t-1 Time t

Tracking as Inference

- $X \in R^n$: the *hidden state* consists of the true parameters (e.g., location, velocity, shape ..)
- $Z \in R^m$: a noisy *measurement of* X
- Bayes rule: $p(x|z) = \frac{p(Z|X)p(X)}{p(Z)}$
- At time t, z_t can be measured
- Our goal: recover most likely state x_t given all noisy observations seen so far

Simple Example

- Known system's linear dynamic model:
  - E.g., $x_t = \begin{pmatrix} p_t \\ \dot{p}_t \end{pmatrix}$,

$\dot{p}$ is velocity

$\dot{p}_{t+1} = \dot{p}_t$ and $p_{t+1} = p_t + \Delta T \dot{p}_t + w_t$

- Known linear mapping between the state x_t and the observation, z_t:
  - E.g., $z_t = p_t + v_t$
- Measurement and estimation errors:
  - v_t and w_t are Gaussian noise

Assumptions

- Known system's linear dynamic model, with white noise $\sim G(0, Q)$

$$\boxed{x_t = Ax_{t-1} + Bu_t + w_{t-1}}$$

  - A is $n \times n$, B is $n \times \ell$
- Known linear transformation of the states to the measurements with white noise $\sim G(0, R)$: $\boxed{z_t = Hx_t + v_t}$
  - H is an $m \times n$ matrix

A, B, H, Q, R are assumed to be known

Goal

$$x_t = Ax_{t-1} + Bu_t + w_{t-1}, \quad w \in G(0, Q)$$
$$z_t = Hx_t + v_t, \quad v_t \in G(0, R)$$

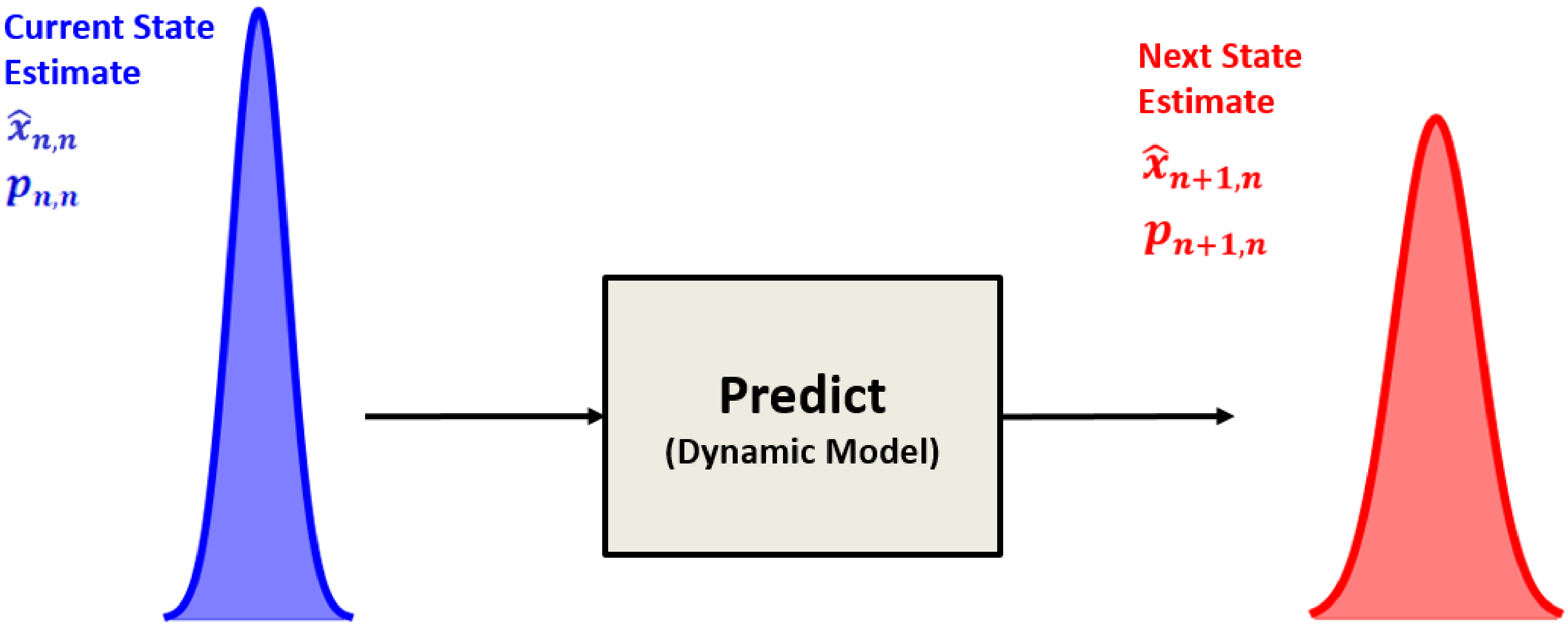
- Compute x_t
- Given:
  - Initial state
  - Previous measurements
  - Known (or learned) linear models: A, B, H
- Minimize the error (its covariance) between the correct and the computed x_t

Notation

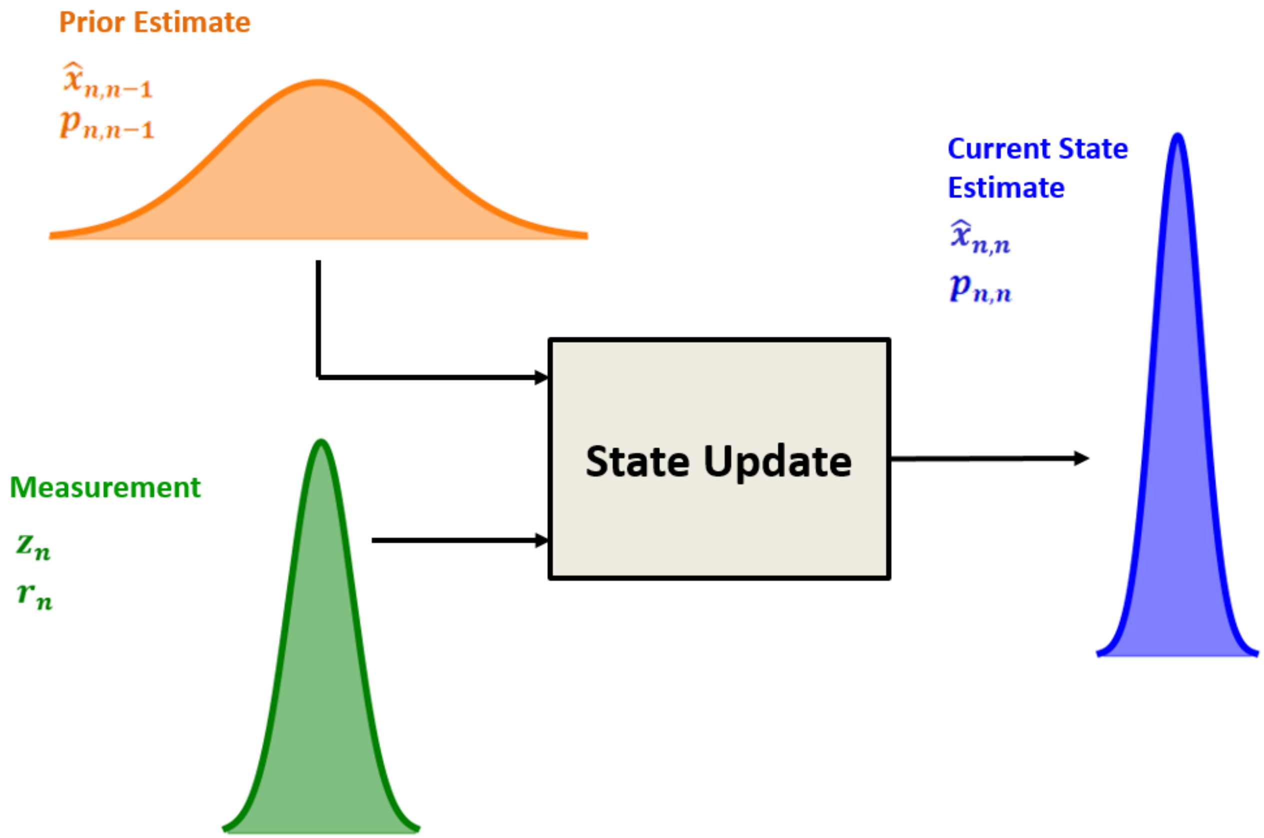
- A priori:
  - state estimation: $\hat{\boldsymbol{x}}_t^-$
  - error: $e_t^- = x_t - \hat{\boldsymbol{x}}_t^-$
  - covariance: $\Sigma_t^- = E(e_t^- \; e_t^{-T})$
- A posteriori, given z_t :
  - state estimation: $\hat{x}_t$
  - error: $e_t = (x_t - \hat{x}_t)$
  - covariance: $\Sigma_t = E(e_t \;\; e_t^T)$

We would like to minimize it

1D


Current State
Estimate
$\hat{x}_{n,n}$
$p_{n,n}$
Predict
(Dynamic Model)
Next State
Estimate
$\hat{x}_{n+1,n}$
$p_{n+1,n}$

1D


Prior Estimate
$\hat{x}_{n,n-1}$
$p_{n,n-1}$
Current State
Estimate
$\hat{x}_{n,n}$
$p_{n,n}$
State Update
Measurement
z_n
r_n

1D

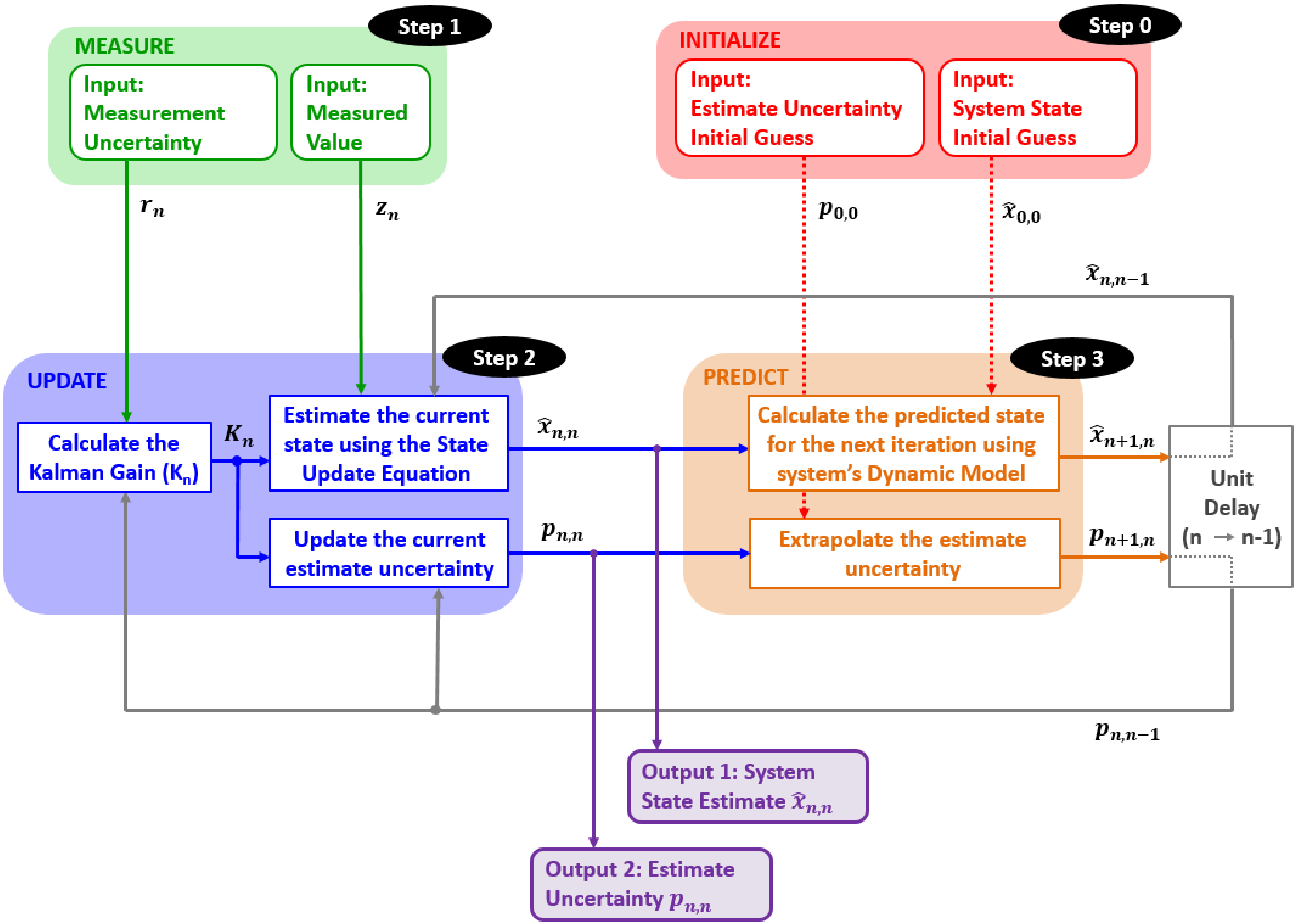
$$K_n = \frac{Variance \quad in \quad Estimate}{Variance \quad in \quad Estimate \quad + \quad Variance \quad in \quad Measurement} = \frac{p_{n,n-1}}{p_{n,n-1} + r_n}$$

Where:

$p_{n,n-1}$ is the extrapolated estimate variance

r_n is the measurement variance

$$p_{n,n} = (1 - K_n)\, p_{n,n-1}$$


MEASURE
Step 1
Input: Measurement Uncertainty
Input: Measured Value
r_n
z_n
INITIALIZE
Step 0
Input: Estimate Uncertainty Initial Guess
Input: System State Initial Guess
$p_{0,0}$
$\hat{x}_{0,0}$
$\hat{x}_{n,n-1}$
UPDATE
Step 2
Calculate the Kalman Gain (K_n)
K_n
Estimate the current state using the State Update Equation
Update the current estimate uncertainty
$\hat{x}_{n,n}$
$p_{n,n}$
PREDICT
Step 3
Calculate the predicted state for the next iteration using system's Dynamic Model
Extrapolate the estimate uncertainty
$\hat{x}_{n+1,n}$
$p_{n+1,n}$
Unit Delay (n → n-1)
$p_{n,n-1}$
Output 1: System State Estimate $\hat{x}_{n,n}$
Output 2: Estimate Uncertainty $p_{n,n}$

1

The Discrete Kalman Filter

Predict

(a priori estimate)

1. Predict the state ahead:

$$\hat{x}_t^- = A\hat{x}_{t-1}^- + Bu$$

2. Predict the error covariance ahead:

$$\Sigma_t^- = A\,\Sigma_{t-1}A^T + Q$$

Update

(a posteriori estimate)

1. Kalman gain K_t is:

$$k_t = \frac{E_{est}}{E_{est} + E_{mea}}$$

$$K_t = \Sigma_t^- H^T \left(H\Sigma_t^- H^T + R\right)^{-1}$$

2. Update the state estimate:

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-)$$

3. Update the error covariance:

$$\Sigma_t = (\mathrm{I} - \mathrm{K}_t\mathrm{H})\Sigma_t^-$$

$$x_t = Ax_{t-1} + Bu_t + w_{t-1}, \quad w \in G(0, Q)$$

$$z_t = Hx_t + v_t, \quad v_t \in G(0, R)$$

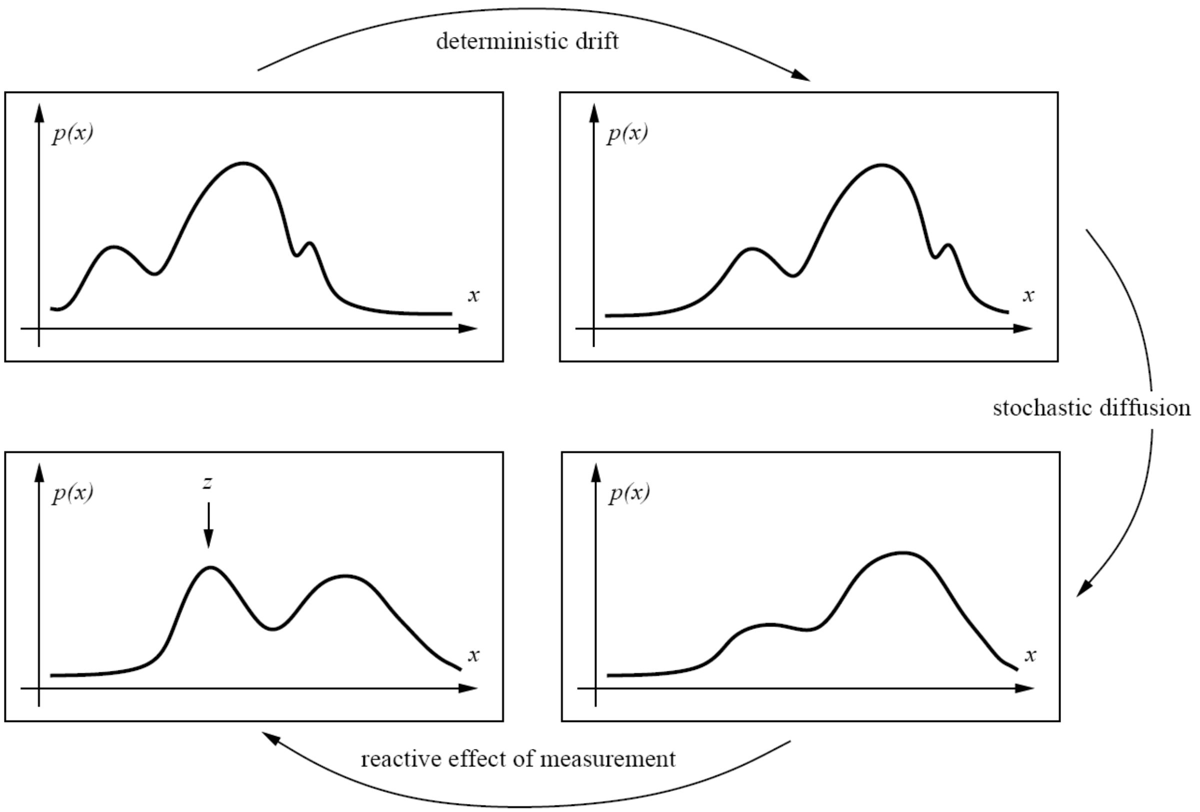
Up to Here

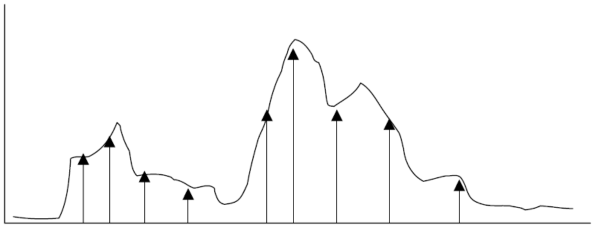
- Up to here: only brief outline of Kalman filter
- Many extensions exists
- Detailed: https://www.kalmanfilter.net/default.aspx
- See more references at the of the presentation

Non-Parametric Prediction

- Motivation: limitations of Kalman filter
- What are they?

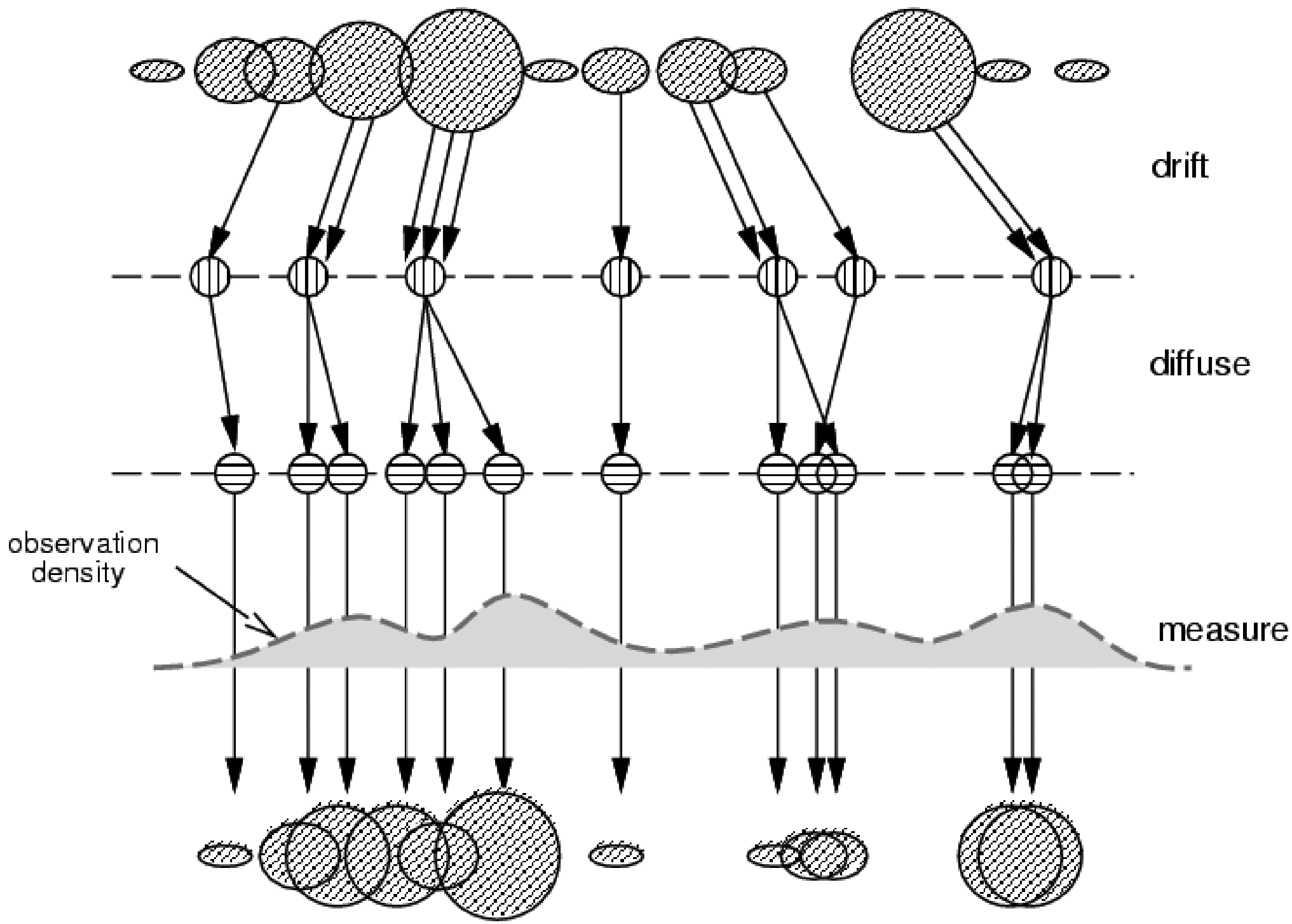
Particle Filters





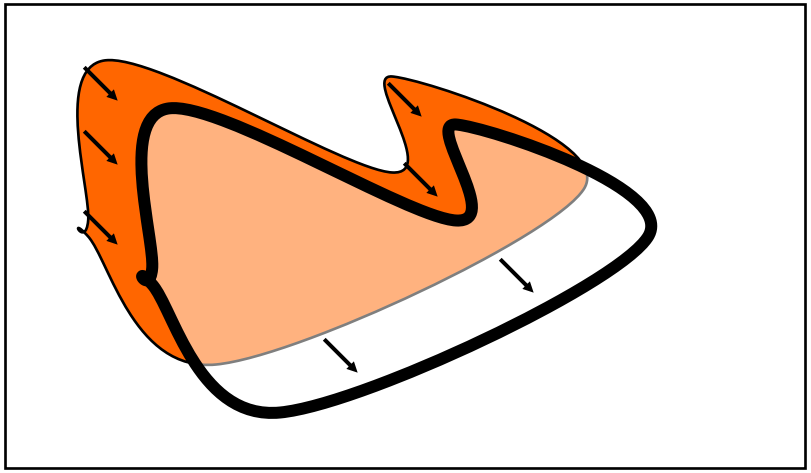
- Can represent distribution with set of weighted samples ("*particles*")
- Allows us to maintain **multiple hypotheses**

Particle Filters



Contour Prediction

- Object model
- Object shape
- Video



Next

- Object recognition / detection
- Course summary

References

- Recent advances and trends in visual tracking: A review, Yang, Ling, Zheng, Wang, Liang & Song, Neurocomputing, 74(18)
- A. Yilmaz, O. Javed, M. Shah. Object tracking: A survey, in: IPCV, 2006.
- CONDENSATION -- conditional density propagation for visual tracking, by Michael Isard and Andrew Blake, Int. J. Computer Vision, 29, 1, 5--28, (1998)
- The original Kalman filter paper: Kalman, R.E. (1960). "A new approach to linear filtering and prediction problems" (PDF). Journal of Basic Engineering
  There are many tutorials on the WEB.
- Video on kalman filter
- https://www.kalmanfilter.net/default.aspx