INF 5300 Advanced Topic: Video Content Analysis

Statistical tracking

Asbjørn Berge

Outline

What is tracking?

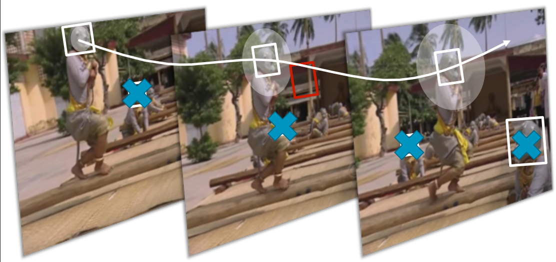
- Definition: using **image measurements** and a **predictive dynamic model** to consistently estimate the state(s) X_t of one or more object(s) over the discrete time steps corresponding to video frames.

What is tracking?

- Why not just do detection?
  - Estimate the state X at each time step
  - - inefficient
  - - data association problem

What is tracking?

- It is better to do tracking
  - Maintain an estimate of X over time, predict the future location
  - + efficient, restricts search space
  - + smoothes noisy measurements
  - - requires knowledge about object behavior



Tracking assumptions

- Smooth camera
  - No instant transitions between viewpoints
  - Any camera pose/parameter changes are gradual
- Object motion can be modeled
  - Linear models
  - Non-linear mod
- Likelihood of object presence at a location in the image can be modeled
  - Typically uses local image information

7

Fundamentals

- Tracking task:
- In the simplest form, tracking can be defined as the problem of estimating the trajectory of an object in the image plane as it moves around a scene. In other words, a tracker assigns consistent labels to the tracked objects in different frames of a video. Additionally, depending on the tracking domain, a tracker can also provide object-centric information, such as orientation, area, or shape of an object.
- Two subtasks:
  - Build some model of what you want to track
  - Use what you know about where the object was in the previous frame(s) to make predictions about the current frame and restrict the search
- Repeat the two subtasks, possibly updating the model

8

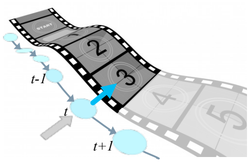
Fundamentals

- Tracking objects can be complex due to:
- loss of information caused by projection of 3D world on 2D image
- noise in images
- complex object shapes / motion
- nonrigid or articulated nature of objects
- partial and full object occlusions
- scene illumination changes
- real-time processing requirements
- Simplify tracking by imposing constraints:
- Almost all tracking algorithms assume that the object motion is smooth with no abrupt changes
- The object motion is assumed to be of constant velocity
- Prior knowledge about the number and the size of objects, or the object appearance and shape

9

Approaches to tracking

- Sequential
  - (recursive, online)
  - + Inexpensive → real-time
  - - no future information
  - - cannot revisit past errors

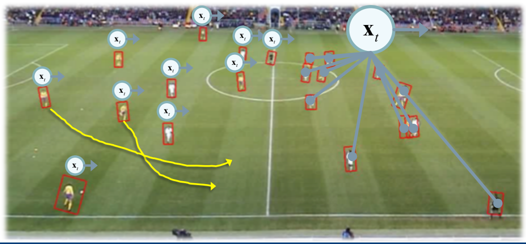


- Batch Processing
  - (offline)
  - - Expensive → not real-time*
  - + considers all information
  - + can correct past errors

10

Approaches to tracking

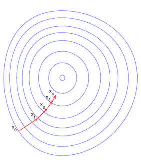
- Parallel trackers
  - several single-object trackers
  - computationally less expensive
  - how to handle interaction, cross-overs?
- Joint state
  - single multi-object representation
  - computationally expensive
  - principled interaction models



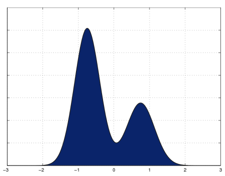
11

Approaches to tracking

- Non-probabilistic
  - + quick convergence*
  - + efficient
  - - stuck in local minima
  - - does not model multiple objects

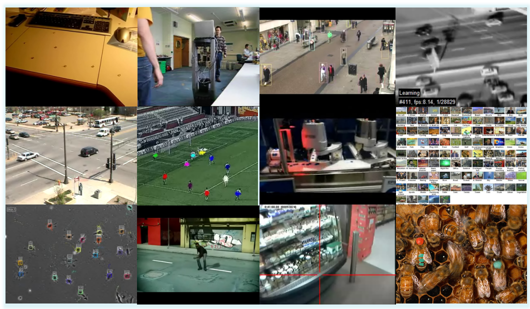


- Probabilistic
  - + flexible, principled
  - + multi-modal
  - - slower
  - - interpretation



12

Tracking applications



13

Tracking applications

- Tracking is an essential step in many computer vision based applications

Detection + Feature Extraction → Tracking → Activity Recognition + Event Recognition → Behavior Analysis + Social Models

Prithwijit Guha , Amitabha Mukerjee and K. S. Venkatesh, Spatio-temporal Discovery: Appearance + Behavior = Agent, Computer Vision, Graphics and Image Processing 4338 516-527, 2007

SINTEF Technology for a better society

14

Tracking Applications

- Sports

P. Nillius, J. Sullivan, S. Carlsson , Multi-Target Tracking - Linking Identities using Bayesian Network Inference
Computer Vision and Pattern Recognition (CVPR), 2006

SINTEF Technology for a better society

15

Tracking Applications

- Surveillance

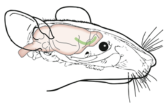
K. Smith, P. Quelhas, and D. Gatica-Perez, Detecting Abandoned Luggage Items in a Public Space, Performance Evaluation of Tracking and Surveillance (PETS) Workshop at CVPR, New York, NY, June 18 2006

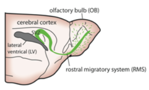
SINTEF Technology for a better society

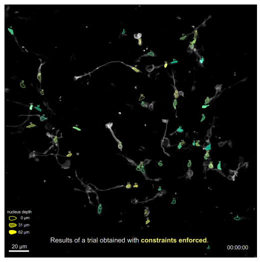
16

Tracking Applications

- Biomed & Microscopy







K. Smith, A. Carleton, and V. Lepetit, General Constraints for Batch Multiple-Target Tracking Applied to Large-Scale Videomicroscopy, Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, June 2008

SINTEF Technology for a better society

17

What is the state of the art?

- Despite being classic computer vision problem, tracking is **largely unsolved**
  - Some limited successes
  - No general-purpose tracker
  - No standard data corpus for comparison
  - No standard evaluation methodology
  - Challenging problems remain

SINTEF Technology for a better society

18

Tracking Challenges

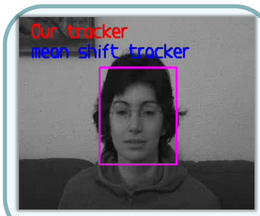
- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Ruei-Sung Lin, David Ross, Jongwoo Lim, Ming-Hsuan Yang, Adaptive discriminative generative model and its Applications Neural Information Processing Systems Conference (NIPS), 2004

SINTEF Technology for a better society

19

Tracking Challenges



- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Amit Adam, Ehud Rivlin and Ilan Shimshoni, Robust Fragments-based Tracking using the Integral Histogram (pdf).
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2006

SINTEF Technology for a better society

20

Tracking Challenges

- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Michael Isard and Andrew Blake CONDENSATION -- conditional density propagation for visual tracking
International Journal of Computer Vision (IJCV), 29, 1, 5--28, (1998)

SINTEF Technology for a better society

21

Tracking Challenges

- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Ruei-Sung Lin, David Ross, Jongwoo Lim, Ming-Hsuan Yang, Adaptive discriminative generative model and its Applications Neural Information Processing Systems Conference (NIPS), 2004

SINTEF Technology for a better society

22

Tracking Challenges

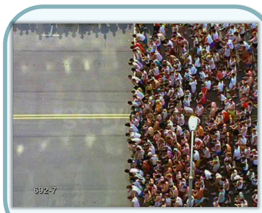
- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Michael Isard and Andrew Blake CONDENSATION -- conditional density propagation for visual tracking
International Journal of Computer Vision (IJCV), 29, 1, 5--28, (1998)

SINTEF Technology for a better society

23

Tracking Challenges



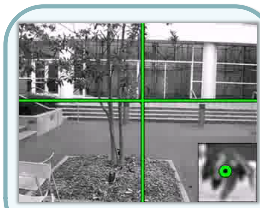
- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Saad Ali and Mubarak Shah, Floor Fields for Tracking in High Density Crowd Scenes, European Conference on Computer Vision (ECCV), 2008.

SINTEF Technology for a better society

24

Tracking Challenges



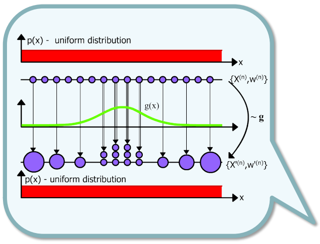
- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- efficient solution

Shawn Lankton, James Malcolm, Arie Nakhmani, and Allen Tannenbaum. Tracking Through Changes in Scale
Proceedings of International Conference on Image Processing (ICIP), 2008.

SINTEF Technology for a better society

25

Tracking Challenges



- appearance change
- occlusion
- distraction
- illumination change
- difficult motion
- multiple objects
- scale change
- finding efficient solution

26

Outline

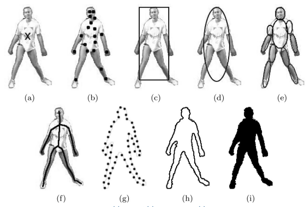
- Introduction to the tracking problem
- Tracking preliminaries
  - Object representations
  - Tracking categories
- Non-probabilistic methods
- Statistical tracking

27

Object Representation

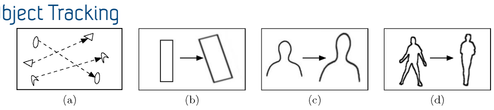
Object representation = Shape + Appearance

Shape representations:



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

Object Tracking



- (a) Point Tracking. Objects detected in consecutive frames are represented by points, and a point matching is done. External mechanism *detect* the objects in every frame.
- (b) Kernel Tracking. Kernel = object shape and appearance. E.g. kernel = a rectangular template or an elliptical shape with an associated histogram. Objects are tracked by computing the motion (parametric transformation such as translation, rotation, and affine) of the kernel in consecutive frames.
- (c)+(d) Silhouette Tracking. Such methods use the information encoded inside the object region (appearance density and shape models). Given the object models, silhouettes are tracked by either shape matching (c) or contour evolution (d). The latter one can be considered as object segmentation applied in the temporal domain using the priors generated from the previous frames. (See lecture on active contours)

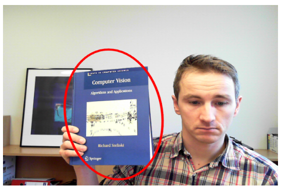
29

Outline

- Introduction to the tracking problem
- Tracking preliminaries
- Non-probabilistic methods
  - Mean shift tracking
  - Lucas-Kanade tracking
  - "Brute-force" – assignments
- Statistical tracking

Mean shift tracking

- The mean-shift algorithm is an efficient approach to tracking objects by appearance
- Ideally, we want an indicator function that returnrs *1* for pixels on the object we are tracking and *0* for all other pixels.
- Not possible - Instead we compute likelihood maps where the value at a pixel is proportional to the likelihood that the pixel comes from the object we are tracking.
- Not limited to only color:
  - edge orientations
  - Texture
  - motion
- Handles occlusions and camouflage poorly

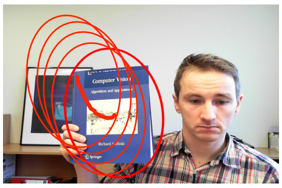


D. Comaniciu, V. Ramesh, and P. Meer, *Kernel-based object tracking*. IEEE Trans. Patt. Analy. Mach. Intell. 25, 564–575, 2003

Mean shift tracking

OpenCV's mean shift tracker implementation uses an algorithm called Camshift. Camshift consists of four steps:

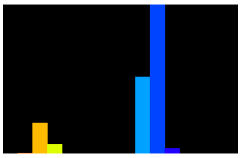
1. Create a color histogram to represent the object
2. Calculate a "object probability" for each pixel in the incoming video frames
3. Shift the location of the *kernel* in each video frame
4. Calculate the size and angle of the ellipse (adapt *kernel*)



Color histogram

Camshift represents the object it's tracking as a histogram of color values in the HSV color model.

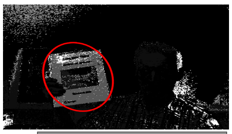
CV book

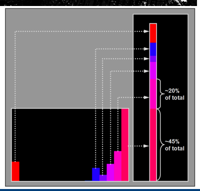


Face

Calculate object probability

- The histogram is in Camshift created only once, at the start of tracking.
- Afterwards, it's used to assign a " object probability" value to each image pixel in the video frames that follow.
- The probability that a pixel selected randomly from the initial region would fall into the rightmost bin is 45%, and so on.
- The hue value for each pixel is thus used to assign a estimated object probability to the pixel.





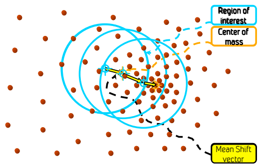
Shift to a new location

Let pixels form a uniform grid of data points, each with a weight (pixel value) proportional to the "likelihood" that the pixel is not on the object we want to track.

"Shifts" the location estimate centered over the area with the highest concentration of bright pixels in the object probability image.

Vector from previous location bycomputing the center of gravity of the probability values within a kernel.

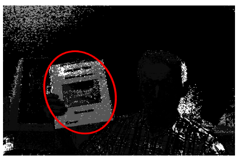
Running mean-shift with kernel ***K*** on a weight image ***w*** is equivalent to performing gradient ascent in a (virtual) image formed by convolving ***w*** with some shadow kernel ***H***



$$\Delta x = \frac{\sum_a K(a-x)\, w(a)\, (a-x)}{\sum_a K(a-x)\, w(a)}$$

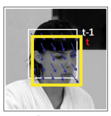
Camshift demo

- "Likelihood" based on color is noisy and is also quantized hard in the example implementation
- Two (three) parameters in Camshift for tweaking the noise sensitivity based on S and V in the HSV color space.
  - Vmin and Vmax : intensity thresholds for colors, low intensity colors are noisy in Hue, and high intensity colors are "too close" to white
  - Smin – saturation threshold. Colors that have low saturation could fit any Hue.



Lucas Kanade Tracking

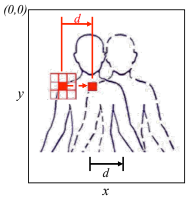
- Traditional Lucas-Kanade is typically run on small, corner-like features (e.g. 5x5) to compute *optical flow*.
- Observation: There's no reason we can't use the same approach on a larger window around the object being tracked.
  - Summarize tracked features
  - Match templates



Sparse motion flow

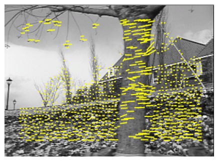
Lucas-Kanade

- Originally intended for fast image registration
- Selects features based on texture:
  - Coefficient matrix based on covariance of image gradients within a window around the proposed feature
  - Eigenvalues of coefficient matrix must be large and similarly valued
- Tracks features based on error:
  - Error between image intensities
  - L_2 Norm (Sum of Squares) used to define error
  - Small changes between frames assumed



Utility of Point Features

Advantages:
- highly repeatable and extensible (work for a variety of images)
- efficient to compute (real time implementations available)
- local methods for processing (tracking through multiple frames)

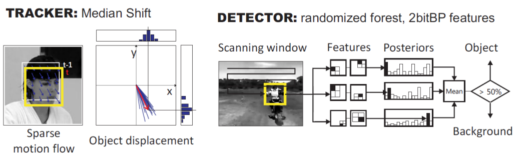


tracking multiple point features = sparse optical flow
sparse point feature tracks yield the image motion

One LK-tracking example; TLD tracker

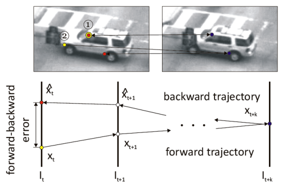
Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-Backward Error: Automatic Detection of Tracking Failures," *International Conference on Pattern Recognition*, 2010, pp. 23-26.

- Continously learns appearance by recalculating a classifier



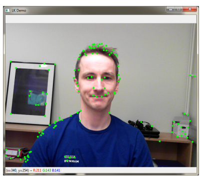
LK tracking: reliability estimates

Selection of reliable points: median is estimated based on 50% of the most reliable points, reliability estimated using combined forward-backward error.
Tracking failure: median residual > threshold.

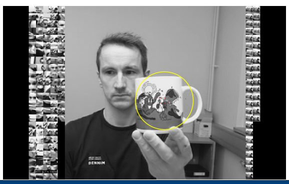


Demo LK-based tracking

- Shi-Tomasi feature tracker

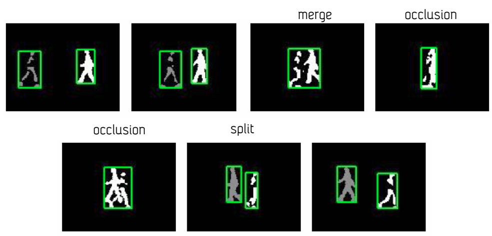


- Continously learning feature tracker



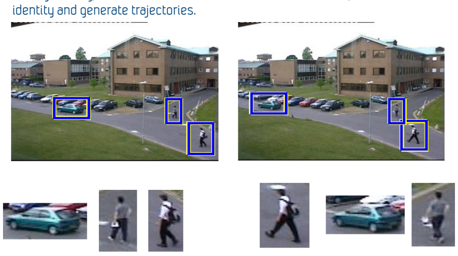
Recall: Blob merge/split

- When two objects pass close to each other, they are detected as a single blob.
- Often, one object will become occluded by the other one. One of the challenging problems is to maintain correct labeling of each object after they split again.



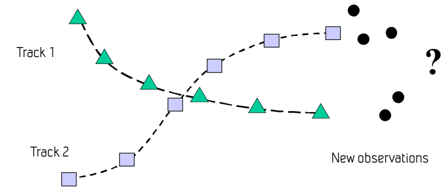
Data association

- More generally, we seek to match a set of blobs across frames, to maintain continuity of identity and generate trajectories.



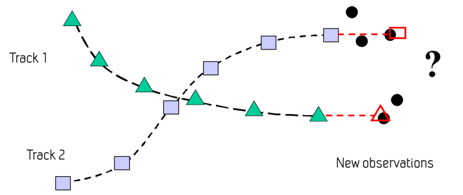
Data Association

- More generally, we seek to match a set of blobs across frames, to maintain continuity of identity and generate trajectories.



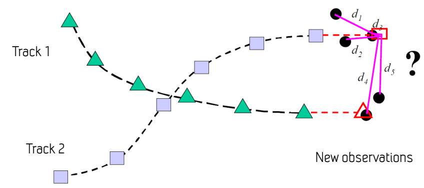
Data Association Scenarios

- Intuition: predict next position along each track



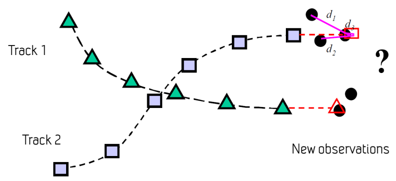
How to determine which observations to add to which track?

- Intuition: predict next position along each track + match should be close to predicted position



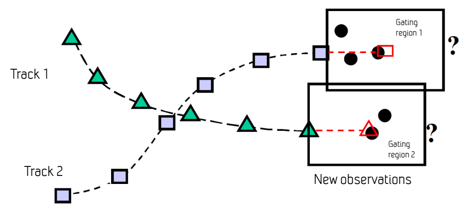
Track matching

- But some matches are fairly unlikely, still



Gating

A method for pruning matches that are geometrically unlikely from the start.
Allows us to decompose matching into smaller subproblems.
Kalman filter – next lecture



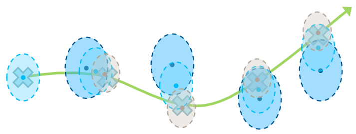
Outline

Recursive Bayesian filtering

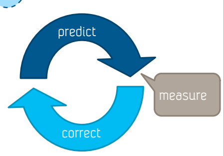
- How is it characterized?
  - Sequential
  - Parallel trackers OR joint modeling of multiple objects
  - Probabilistic
- Popular examples
  - Kalman filter
  - Particle filter

Recursive Bayesian Filtering

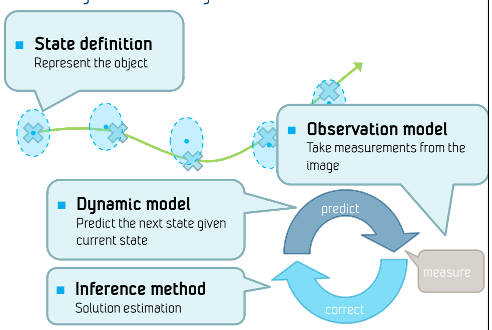
- **Key idea 1.** PDFs represent our belief as to the state of the object



- **Key idea 2.** recursive cycle
  - **Predict** from motion model
  - **Measurement** from image
  - **Correct** the prediction



Recursive Bayesian Filtering



Tracking ingredients

- State Definition

- Observation Model

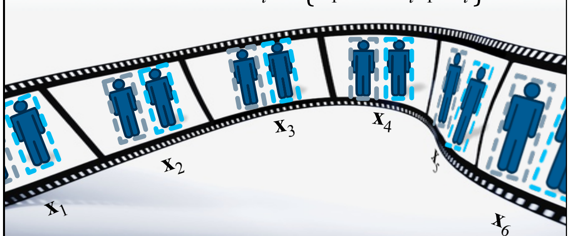
- Dynamic Model

- Inference

State definition

- Describes properties of the tracked object(s) at an instant in time
- Defines solution space

$$X_t = \{\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{x}_t\}$$

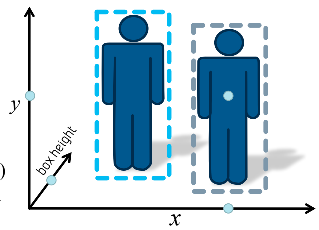


55

State definition

- Decomposed for time step t, $\mathbf{X}_t$ can parameterize the object in many ways, often via:
  - location
  - velocity
  - size
  - shape
  - identity
  - switching model

$$\mathbf{x}_t = x$$
$$\mathbf{x}_t = (x, y)$$
$$\mathbf{x}_t = (x, y, h)$$
$$\mathbf{x}_t = \{\mathbf{x}_t^1, \mathbf{x}_t^2\}$$



SINTEF Technology for a better society

56

State definition

- Object defined by a point
  - position
  - velocity
  - acceleration

$$\mathbf{x}_t = (x, y)$$
$$\mathbf{x}_t = (x, y, w, h)$$
$$\mathbf{x}_t = (x, y, \dot{x}, \dot{y}, w, h)$$

Saad Ali and Mubarak Shah, Floor Fields for Tracking in High Density Crowd Scenes, European Conference on Computer Vision (ECCV), 2008.

SINTEF Technology for a better society

57

State definition

- Bounding box
  - position
  - height
  - aspect
  - velocity

$$\mathbf{x}_t = (x, y)$$
$$\mathbf{x}_t = (x, y, h, a)$$
$$\mathbf{x}_t = (x, y, \dot{x}, \dot{y}, h, a)$$

M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Robust Tracking-by-Detection using a Detector Confidence Particle Filter, International Conference on Computer Vision (ICCV), 2009

SINTEF Technology for a better society

58

State definition

- Ellipse
  - location
  - eccentricity
  - major axis

$$\mathbf{x}_t = (x, y, m, e)$$
$$\mathbf{x}_t = (x, y, a, b)$$

Yuan Li, Chang Huang, Ram Nevatia Learning to Associate: Hybrid Boosted Multi-Target Tracker for Crowded Scene, Computer Vision and Pattern Recognition (CVPR), June 2009

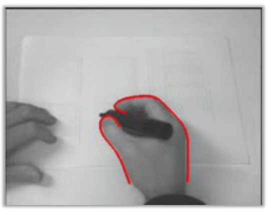
SINTEF Technology for a better society

59

State definition

- Active contour
  - b-splines
  - control points
  - spline length
  - H basis functions

$$\mathbf{x}_t = (X(s), Y(s))$$
$$X(s) = H(s)\mathrm{X},\ 0 \le s \le N$$
$$Y(s) = H(s)\mathrm{Y}$$
$$\mathrm{X} = \{x^1, x^s, \ldots, x^N\}$$
$$\mathrm{Y} = \{y^1, y^s, \ldots, y^N\}$$



Michael Isard and Andrew Blake CONDENSATION -- conditional density propagation for visual tracking International Journal of Computer Vision (IJCV), 29, 1, 5--28, (1998)

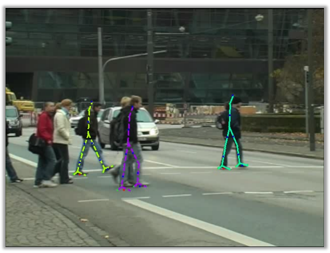
SINTEF Technology for a better society

60

State definition

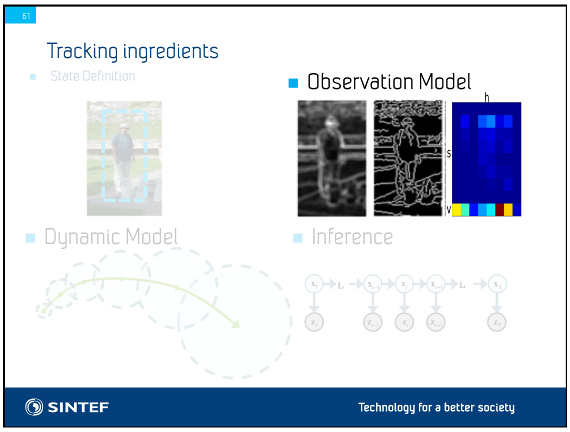
- Articulated & Part-based Models
  - set of vertices
  - locations
  - scales
  - constraints

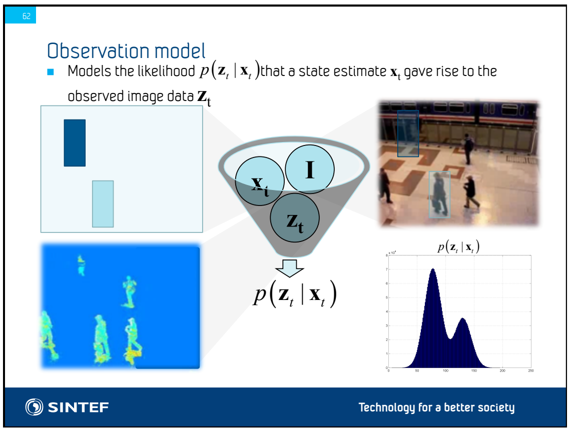
$$\mathbf{x}_t = \{v^1, v^i, \ldots, v^N\}$$
$$v^i = (x^i, y^i, s^i)$$

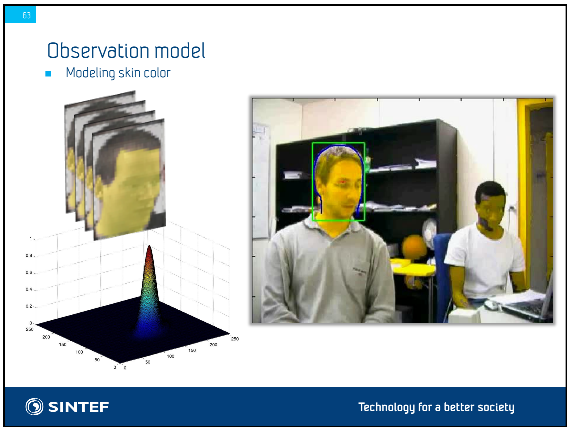


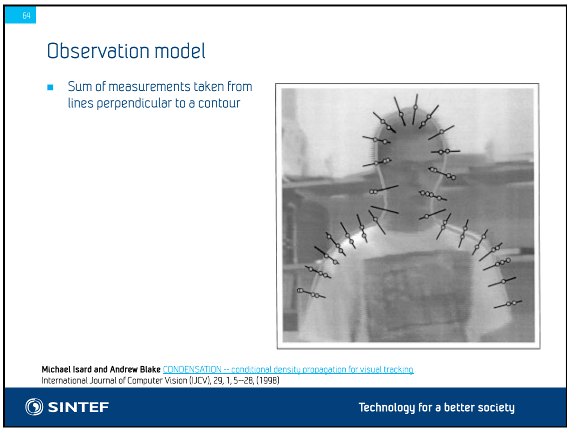
M. Andriluka, S. Roth, B. Schiele, People Tracking by Detection and People-Detection-by-Tracking, *Computer Vision and Pattern Recognition (CVPR'08), Anchorage, USA, June 2008*

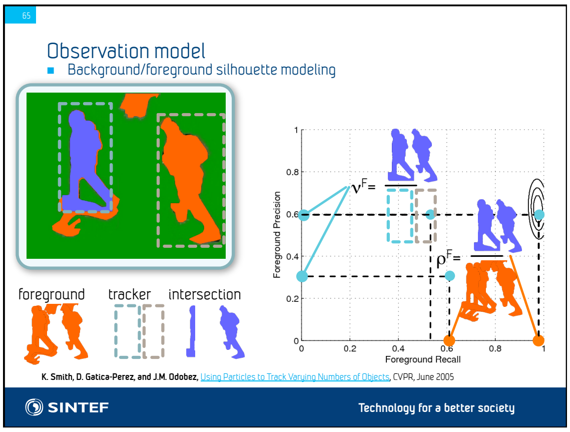
SINTEF Technology for a better society

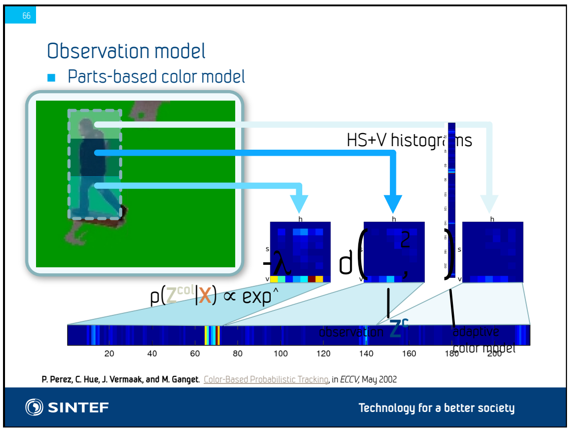

61
Tracking ingredients
State Definition
Observation Model
Dynamic Model
Inference
SINTEF
Technology for a better society


62
Observation model
Models the likelihood $p(\mathbf{z}_t \mid \mathbf{x}_t)$ that a state estimate $\mathbf{x}_t$ gave rise to the observed image data $\mathbf{Z}_t$
$p(\mathbf{z}_t \mid \mathbf{x}_t)$
SINTEF
Technology for a better society


63
Observation model
Modeling skin color
SINTEF
Technology for a better society


64
Observation model
Sum of measurements taken from lines perpendicular to a contour
Michael Isard and Andrew Blake CONDENSATION -- conditional density propagation for visual tracking
International Journal of Computer Vision (IJCV), 29, 1, 5–28, (1998)
SINTEF
Technology for a better society

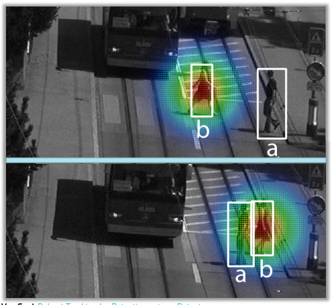

65
Observation model
Background/foreground silhouette modeling
foreground
tracker
intersection
Foreground Precision
Foreground Recall
K. Smith, D. Gatica-Perez, and J.M. Odobez, Using Particles to Track Varying Numbers of Objects, CVPR, June 2005
SINTEF
Technology for a better society


66
Observation model
Parts-based color model
HS+V histograms
P. Perez, C. Hue, J. Vermaak, and M. Gangnet Color-Based Probabilistic Tracking, in ECCV, May 2002
SINTEF
Technology for a better society

67

Observation model

- Detector confidence
  - HOG based sliding window detector



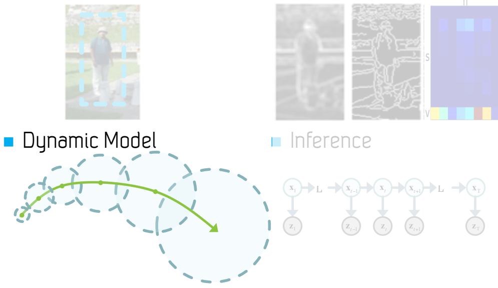
M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Robust Tracking-by-Detection using a Detector Confidence Particle Filter, International Conference on Computer Vision (ICCV), 2009

SINTEF Technology for a better society

68

Tracking ingredients

- State Definition
- Observation Model
- Dynamic Model
- Inference



SINTEF Technology for a better society

69

Dynamic model

- Current state is predicted from previous state

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = N\left(\mathbf{F}_t \mathbf{x}_{t-1}, \Sigma_{F_t}\right)$$

$x_t \sim N(F_t x_{t-1}, \Sigma_{F_t})$ to obtain samples

- Autoregressive linear dynamic model

$$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{w}_t$$

Predicted state time t

State transition model

Previous state time $t-1$

Noise term $\mathbf{w}_t \sim N(0,\mathbf{Q}_t)$

SINTEF Technology for a better society

70

Dynamic model

- 1st order autoregressive
  - models position & velocity

$$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{w}_t$$

State vector

$$\mathbf{x}_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix} \begin{pmatrix} x_t \\ y_t \\ \dot{x}_t \\ \dot{y}_t \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & & 1 & \\ & 1 & & 1 \\ & & 1 & \\ & & & 1 \end{pmatrix}}_{\mathbf{F}_t} \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ \dot{x}_{t-1} \\ \dot{y}_{t-1} \end{pmatrix} + \mathbf{w}_t$$

State transition model

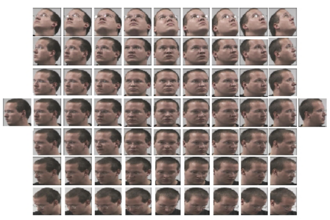
Previous state

SINTEF Technology for a better society

71

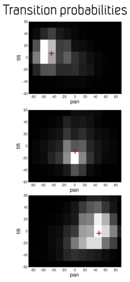
Dynamic model

- Nonlinear dynamic models
  - Discrete state transitions



Discrete pose states

Transition probabilities

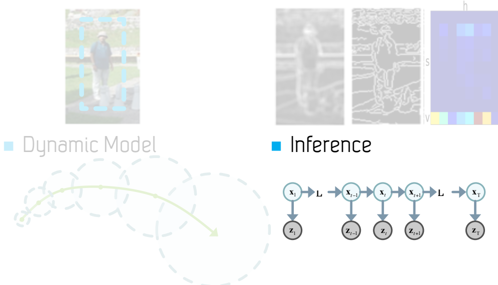


SINTEF Technology for a better society

72

Tracking ingredients

- State Definition
- Observation Model
- Dynamic Model
- Inference



SINTEF Technology for a better society

73

Recursive Bayesian filtering

- Filtering equation:

$$\underbrace{p(\mathbf{x}_t \mid Z_t)}_{\text{posterior estimate}} \propto \underbrace{p(\mathbf{z}_t \mid \mathbf{x}_t)}_{\text{likelihood or observation model}} \int_{\mathbf{x}_{t-1}} \underbrace{p(\mathbf{x}_t \mid \mathbf{x}_{t-1})}_{\text{motion model or dynamic model}} \underbrace{p(\mathbf{x}_{t-1} \mid Z_{t-1})}_{\text{posterior estimated at t-1}}$$

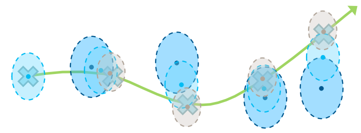
- Definitions
  - State from 1 to time t: $X_t = \{\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{x}_t\}$
  - Observations from 1 to time t: $Z_t = \{\mathbf{z}_1, \ldots, \mathbf{z}_{t-1}, \mathbf{z}_t\}$

SINTEF Technology for a better society

74

Recursive Bayesian filtering

- Use probability distributions to model the tracking problem



$$\underbrace{p(\mathbf{x}_t \mid Z_t)}_{\text{posterior}} \propto \underbrace{p(\mathbf{z}_t \mid \mathbf{x}_t)}_{\text{likelihood}} \int_{\mathbf{x}_{t-1}} \underbrace{p(\mathbf{x}_t \mid \mathbf{x}_{t-1})}_{\text{motion model}} \underbrace{p(\mathbf{x}_{t-1} \mid Z_{t-1})}_{\text{posterior at } t\text{-}1}$$

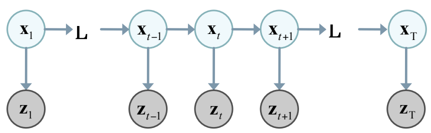
correct measure predict

SINTEF Technology for a better society

75

Modeling the tracking problem

- Model the problem as a Hidden Markov Model (HMM)
  - Dependency
  - Variables: hidden observed



- Assumptions
  - Dynamics form a Markov chain $p(\mathbf{x}_t \mid X_{t-1}) = p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$
  - Independent observations $p(\mathbf{z}_t \mid \mathbf{x}_t, X_{t-1}, Z_{t-1}) = p(\mathbf{z}_t \mid \mathbf{x}_t)$

SINTEF Technology for a better society

76

Recursive Bayesian filtering

Derivation setup

- Notation
  - State from 0 to time t: $X_t = \{\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, \mathbf{x}_t\}$
  - Observations from 0 to time t: $Z_t = \{\mathbf{z}_1, \ldots, \mathbf{z}_{t-1}, \mathbf{z}_t\}$
- Assumptions
  - Dynamics form a Markov chain $p(\mathbf{x}_t \mid X_{t-1}) = p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$
  - Independent observations $p(\mathbf{z}_t \mid \mathbf{x}_t, X_{t-1}, Z_{t-1}) = p(\mathbf{z}_t \mid \mathbf{x}_t)$

SINTEF Technology for a better society

77

Recursive Bayesian filtering

Derivation setup

starting from this relation

$$p(X_t, Z_t) = p(\mathbf{x}_t, \mathbf{z}_t, X_{t-1}, Z_{t-1})$$

derive the recursive Bayesian filtering equation

$$p(\mathbf{x}_t \mid Z_t) \propto p(\mathbf{z}_t \mid \mathbf{x}_t) \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid Z_{t-1})$$

SINTEF Technology for a better society

78

Useful probability relations

$$P(A \cap B) = P(A, B) = P(A \text{ and } B)$$
$$P(A \cup B) = P(A \text{ or } B)$$

- Conditional probability

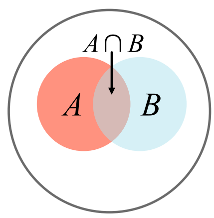
$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$
$$P(B \mid A) = \frac{P(A \cap B)}{P(A)}$$

- Bayes theorem

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

- Marginal probability

$$P(A) = \sum_B P(A, B) = \sum_B P(A \mid B)P(B)$$
$$P(A) = \int_B P(A, B) = \int_B P(A \mid B)P(B)$$



SINTEF Technology for a better society

79

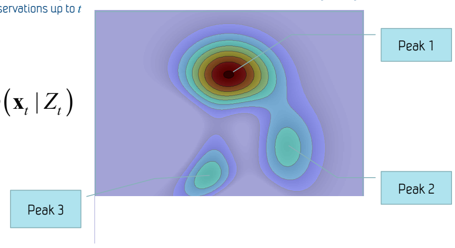
Probability distribution to model belief in object location

- Tracking faces in frame *t*

80

Probability distribution to model belief in object location

- *Posterior* or *target* distribution – models belief as to the state of the system given the observations up to *t*
- $p(\mathbf{x}_t \mid Z_t)$

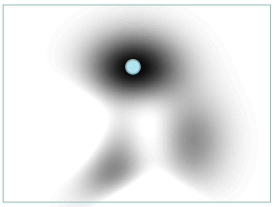


SINTEF Technology for a better society

81

Representing the posterior

- A point (dirac) $p(\mathbf{x}_t \mid Z_t) = \begin{cases} 1 & \text{if } \mathbf{x}_t = \mu \\ 0 & otherwise \end{cases}$

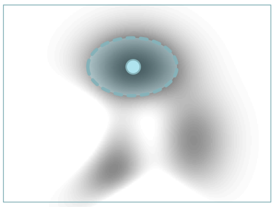


SINTEF Technology for a better society

82

Representing the posterior

- Gaussian $p(\mathbf{x}_t \mid Z_t) = N(\mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n \mid \Sigma \mid}} \exp\left(-\frac{1}{2}(\mathbf{x}_t - \mu)^{\mathrm{T}} \Sigma^{-1} (\mathbf{x}_t - \mu)\right)$

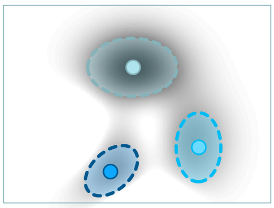


SINTEF Technology for a better society

83

Representing the posterior

- Mixture of Gaussians $\{(\mu_1, \Sigma_1), (\mu_2, \Sigma_2), \ldots\}$



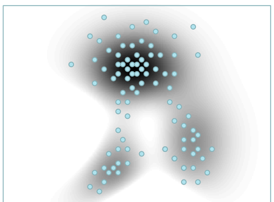
$$p(\mathbf{x}_t \mid Z_t) \propto \sum_i \frac{1}{\sqrt{(2\pi)^n \mid \Sigma_i \mid}} \exp\left(-\frac{1}{2}(\mathbf{x}_t - \mu_i)^{\mathrm{T}} \Sigma_i^{-1} (\mathbf{x}_t - \mu_i)\right)$$

SINTEF Technology for a better society

84

Representing the posterior

- Set of discrete samples (particles) $\{x_t^{(n)}, n = 1, \ldots, N\}$



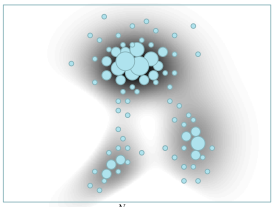
$$p(\mathbf{x}_t \mid Z_t) \approx \sum_{n=1}^{N} \delta(x_t - x_t^{(n)})$$

SINTEF Technology for a better society

85

Representing the posterior

- Set of weighted samples (particles) $\left\{x_t^{(n)}, w_t^{(n)}\right\}_{n=1}^{N}$



$$w_t^{(n)} \in [0,1]$$

$$\sum_n w_t^{(n)} = 1$$

$$p(\mathbf{x}_t \mid Z_t) \approx \sum_{n=1}^{N} w_t^{(n)}\, \delta\left(x_t - x_t^{(n)}\right)$$

SINTEF Technology for a better society

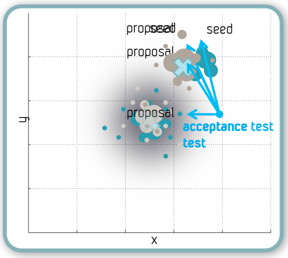
Recursive Bayesian Filtering

- Model belief about the current state X_t given past and present observed data $Z_{1:t}$

Kalman filter exact solution
[1] Kalman, R.E. *A new approach to linear filtering and prediction problems.* ASME, Journal of Basic Engineering, 1960.

SIR particle filter discrete approx.
[2] M. Isard and A. Blake. *Condensation*, International Journal of Computer Vision, 1998.

MCMC particle filter discrete approx.
[3] Z. Khan, T. Balch, and F. Dellaert, *An MCMC-based particle filter for tracking multiple interacting targets*, ECCV, 2004.

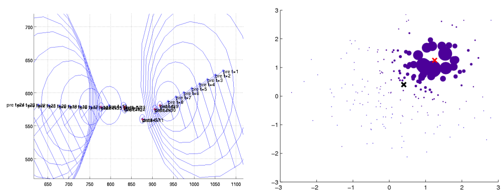


SINTEF Technology for a better society

87

Recursive bayesian filtering

- Kalman filter
  exact solution
  - Continuous state space
  - Linear dynamics
  - Gaussian observation density
- Particle filter
  approximate solution
  - Continuous, discrete, or mixed state space
  - Arbitrary dynamics
  - Arbitrary observation density

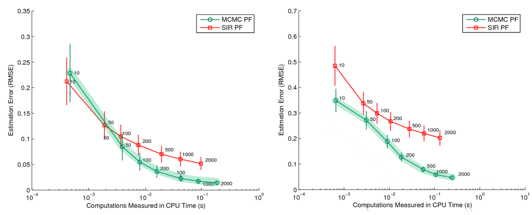


SINTEF Technology for a better society

88

MCMC vs SIR

- task: track **1D targets** using **SIR PF** & **MCMC PF**



10 targets **50 targets**

SINTEF Technology for a better society

89

Kalman filter

- Published in 1960
  Kalman, R. E. 1960. "A New Approach to Linear Filtering and Prediction Problems," Transaction of the ASME—Journal of Basic Engineering, pp. 35-45 (March 1960).
- Used for many problems
  - Guidance
  - Navigation
  - Autopilots
  - Radar
  - Satellite
  - Weather forecasting

SINTEF Technology for a better society

90

Kalman filter: Gaussians!

- in bayesian filtering terms
  - Posterior

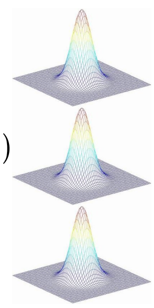
$$p(\mathbf{x}_t \mid Z_t) = N\left(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}\right)$$

  - motion model

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = N\left(\mathbf{F}_t \mathbf{x}_{t-1}, \mathbf{Q}_t\right)$$

  - observation model

$$p(\mathbf{z}_t \mid \mathbf{x}_t) = N\left(\mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t\right)$$

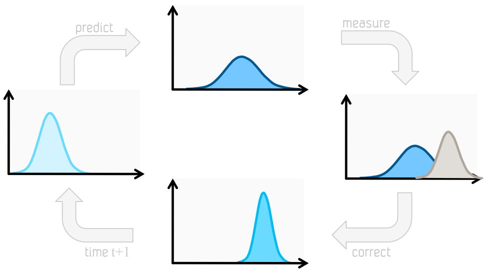


SINTEF Technology for a better society

91

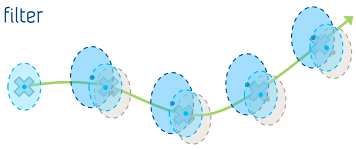
Probability density propagation

- Gaussian → Kalman filter

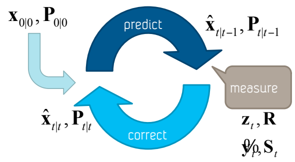


92

Kalman filter

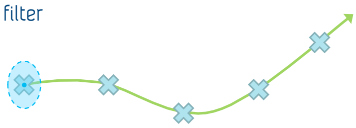


- Predict, measure, correct cycle iteratively estimates the state at each time step



93

Kalman filter

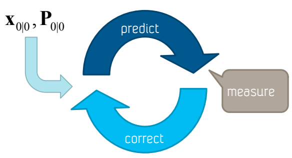


- State vector

$$\mathbf{x}_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix}$$

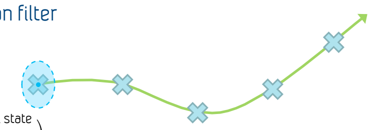
- Measurement

$$\mathbf{z}_t = \begin{pmatrix} x \\ y \end{pmatrix}$$



94

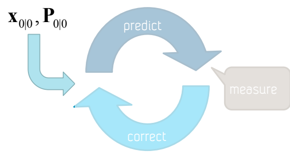
Kalman filter



- Initial state

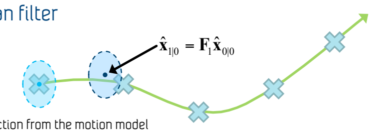
$$\mathbf{x}_{0|0} = \begin{pmatrix} x_0 \\ y_0 \\ \dot{x}_0 \\ \dot{y}_0 \end{pmatrix}$$

$$\mathbf{P}_{0|0} = \begin{pmatrix} L & & & \\ & L & & \\ & & L & \\ & & & L \end{pmatrix}$$



95

Kalman filter

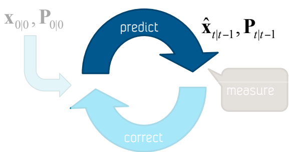


- Prediction from the motion model

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = N(\mathbf{F}_t \mathbf{x}_{t-1}, \mathbf{Q}_t)$$

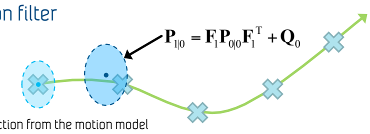
- Update the mean

$$\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1}$$



96

Kalman filter

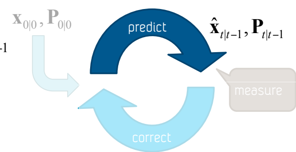


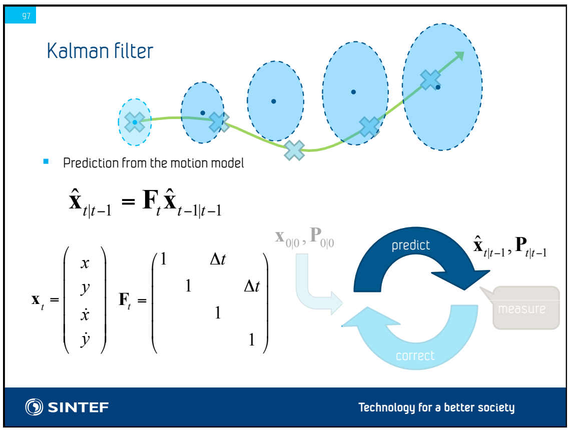
- Prediction from the motion model

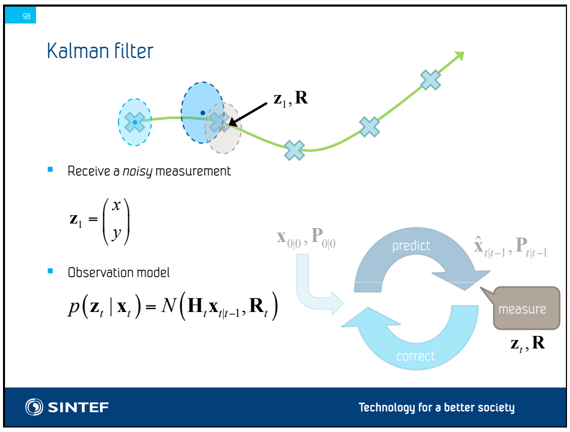
$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = N(\mathbf{F}_t \mathbf{x}_{t-1}, \mathbf{Q}_t)$$

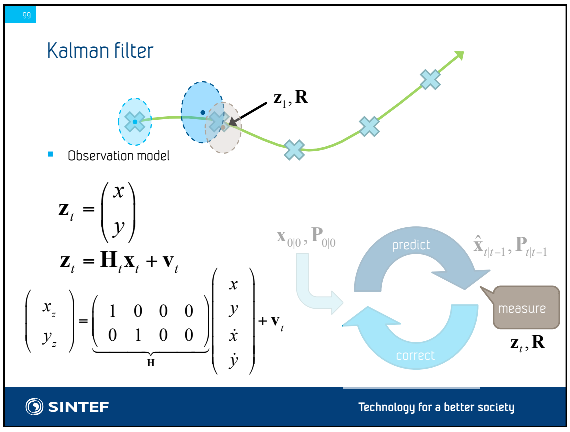
- Update covariance

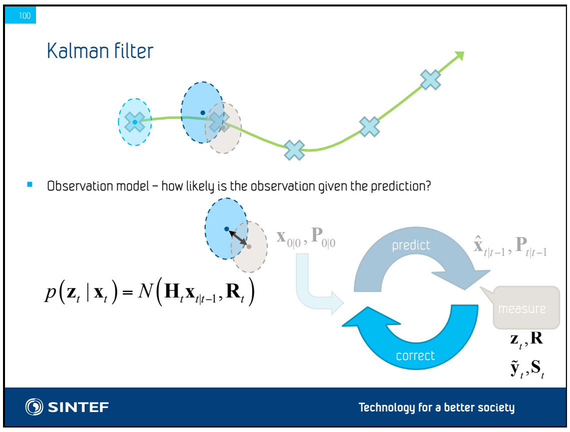
$$\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^{\mathrm{T}} + \mathbf{Q}_{t-1}$$

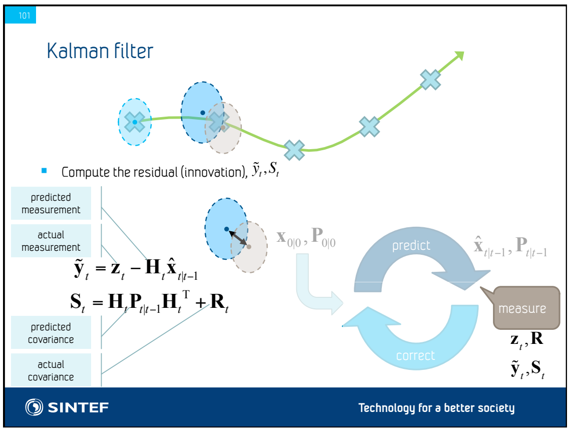


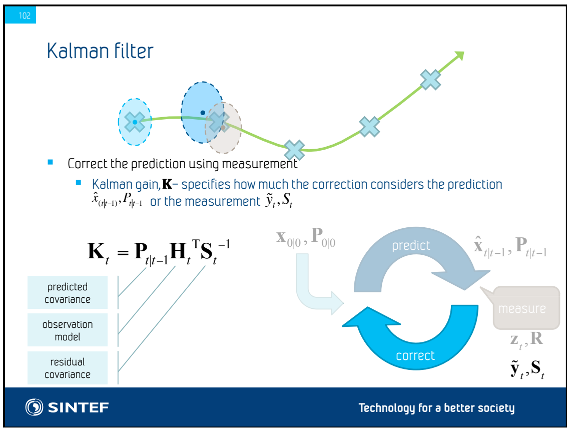

97
Kalman filter
Prediction from the motion model
$\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1}$
$\mathbf{x}_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix}$ $\mathbf{F}_t = \begin{pmatrix} 1 & & \Delta t & \\ & 1 & & \Delta t \\ & & 1 & \\ & & & 1 \end{pmatrix}$
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
correct
SINTEF
Technology for a better society


98
Kalman filter
$\mathbf{z}_1, \mathbf{R}$
Receive a *noisy* measurement
$\mathbf{z}_1 = \begin{pmatrix} x \\ y \end{pmatrix}$
Observation model
$p(\mathbf{z}_t \mid \mathbf{x}_t) = N(\mathbf{H}_t \mathbf{x}_{t|t-1}, \mathbf{R}_t)$
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
$\mathbf{z}_t, \mathbf{R}$
correct
SINTEF
Technology for a better society


99
Kalman filter
$\mathbf{z}_1, \mathbf{R}$
Observation model
$\mathbf{z}_t = \begin{pmatrix} x \\ y \end{pmatrix}$
$\mathbf{z}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{v}_t$
$\begin{pmatrix} x_z \\ y_z \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}_{\mathbf{H}} \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix} + \mathbf{v}_t$
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
$\mathbf{z}_t, \mathbf{R}$
correct
SINTEF
Technology for a better society

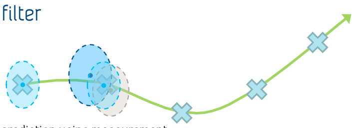

100
Kalman filter
Observation model – how likely is the observation given the prediction?
$p(\mathbf{z}_t \mid \mathbf{x}_t) = N(\mathbf{H}_t \mathbf{x}_{t|t-1}, \mathbf{R}_t)$
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
$\mathbf{z}_t, \mathbf{R}$
$\tilde{\mathbf{y}}_t, \mathbf{S}_t$
correct
SINTEF
Technology for a better society


101
Kalman filter
Compute the residual (innovation), $\tilde{y}_t, S_t$
predicted measurement
actual measurement
$\tilde{\mathbf{y}}_t = \mathbf{z}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}$
$\mathbf{S}_t = \mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^{\mathrm{T}} + \mathbf{R}_t$
predicted covariance
actual covariance
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
$\mathbf{z}_t, \mathbf{R}$
$\tilde{\mathbf{y}}_t, \mathbf{S}_t$
correct
SINTEF
Technology for a better society


102
Kalman filter
Correct the prediction using measurement
Kalman gain, $\mathbf{K}$ – specifies how much the correction considers the prediction $\hat{x}_{(t|t-1)}, P_{t|t-1}$ or the measurement $\tilde{y}_t, S_t$
$\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^{\mathrm{T}} \mathbf{S}_t^{-1}$
predicted covariance
observation model
residual covariance
$\mathbf{x}_{0|0}, \mathbf{P}_{0|0}$
predict
$\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}$
measure
$\mathbf{z}_t, \mathbf{R}$
$\tilde{\mathbf{y}}_t, \mathbf{S}_t$
correct
SINTEF
Technology for a better society

103

Kalman filter



- Correct the prediction using measurement

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t \tilde{\mathbf{y}}_k$$

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t \left(\mathbf{z}_t - \mathbf{H}\hat{\mathbf{x}}_{t|t-1}\right)$$

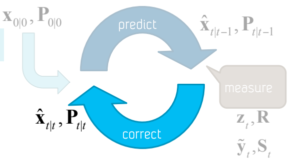
state prediction

residual

$$\mathbf{P}_{t|t} = \left(I - \mathbf{K}_t \mathbf{H}_t\right)\mathbf{P}_{t|t-1}$$

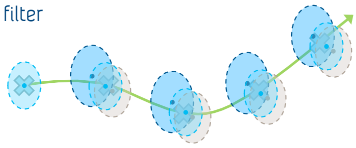
predicted covariance

residual covariance

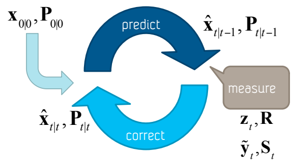


104

Kalman filter



- Predict, measure, correct cycle iteratively estimates the state at each time step



105

Kalman filter

Kalman filter smoothing of accelerometer measurements.

106

Kalman filter

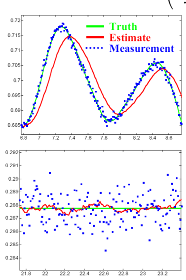
Kalman filter tracking an aircraft.

Kalman filter tracking an aircraft.

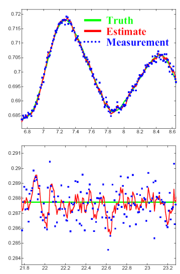
107

Kalman filter: dynamic models

- Without velocity $\mathbf{x}_t = \begin{pmatrix} x \\ y \end{pmatrix}$



- Constant velocity model $\mathbf{x}_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix}$



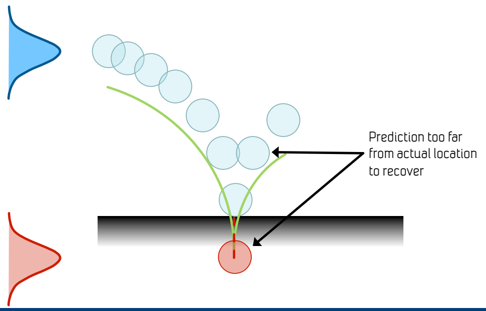
108

Summary: Kalman filter

- **Pros +**
  - Gaussian densities easy to work with
  - Simple updates, compact & efficient
  - Well established method
- **Cons -**
  - Restricted to Gaussian densities
  - Unimodal distribution: single hypothesis
  - Dynamic model restricted to linear, continuous

109

Kalman limitations

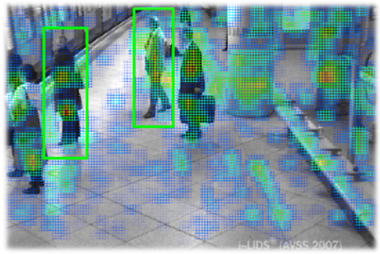


SINTEF Technology for a better society

110

Kalman limitations

- Cannot use non-Gaussian observation models

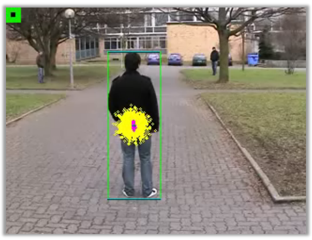


M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool,
Robust Tracking-by-Detection using a Detector Confidence Particle Filter , International Conference on Computer Vision (ICCV), 2009

SINTEF Technology for a better society

111

Particle filter



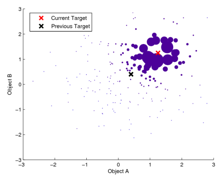
D. Klein, D. Schulz, S. Frintrop, and A. Cremers, Adaptive Real-Time Video Tracking for Arbitrary Objects ,
International Conference on Intelligent Robots and Systems (IROS), 2010

SINTEF Technology for a better society

112

Particle filters

- Go by many names:
  - Sequential Monte Carlo Methods
  - Sequential importance resampling (SIR)
  - Bootstrap filters
  - Condensation trackers
- Originally used for problems in
  - Statistics
  - Fluid mechanics
  - Statistical mechanics
  - Signal processing
- A lot of tools already developed in the other disciplines
- Introduced to computer vision community by
Michael Isard and Andrew Blake, *CONDENSATION -- Conditional Density Propagation for Visual Tracking*, International Journal of Computer Vision (IJCV), 29, 1, 5--28, (1998)

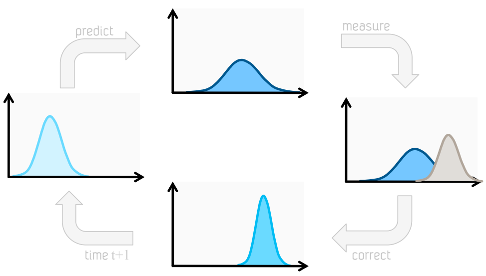


SINTEF Technology for a better society

113

Probability density propagation

- **Gaussian densities** → Kalman filter

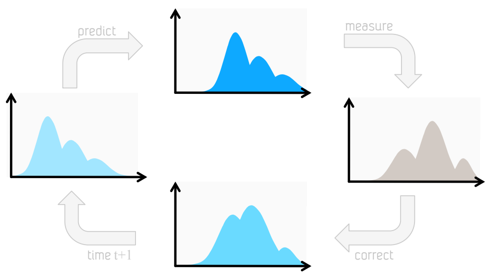


SINTEF Technology for a better society

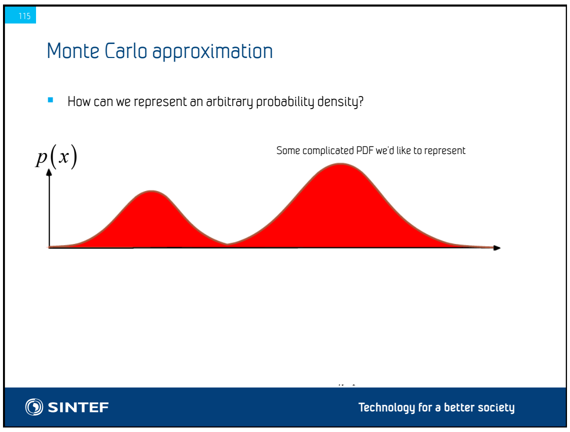
114

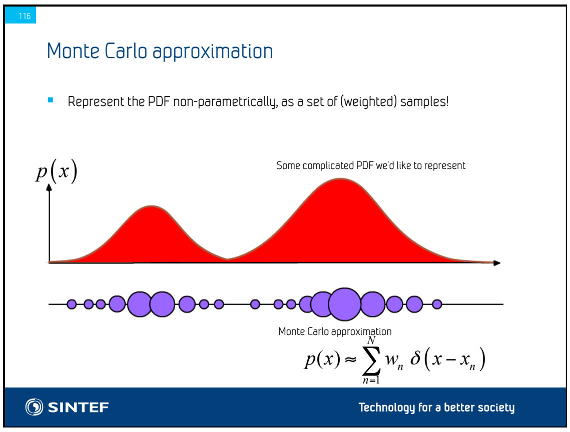
Probability density propagation

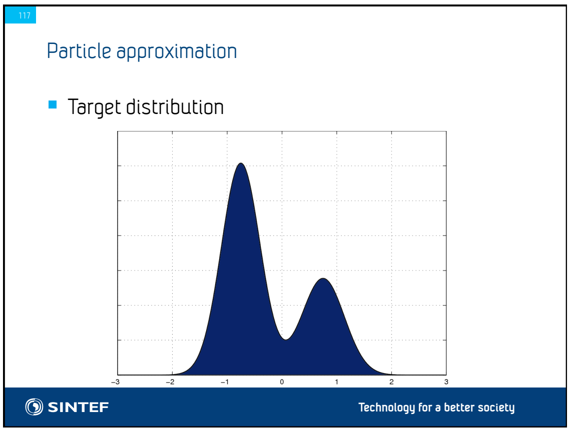
- **General densities** → particle filter

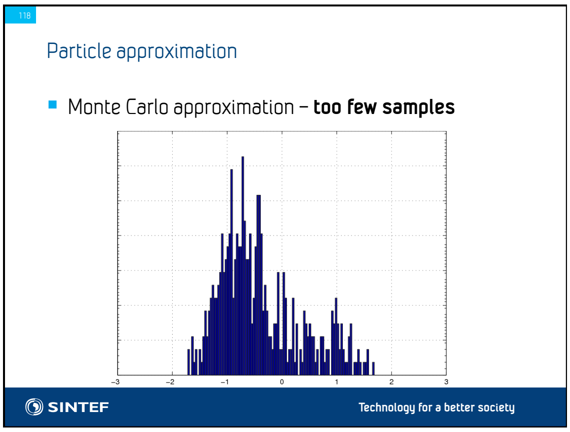


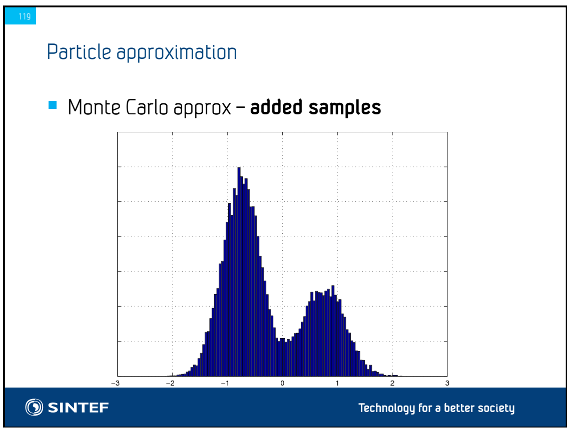
SINTEF Technology for a better society

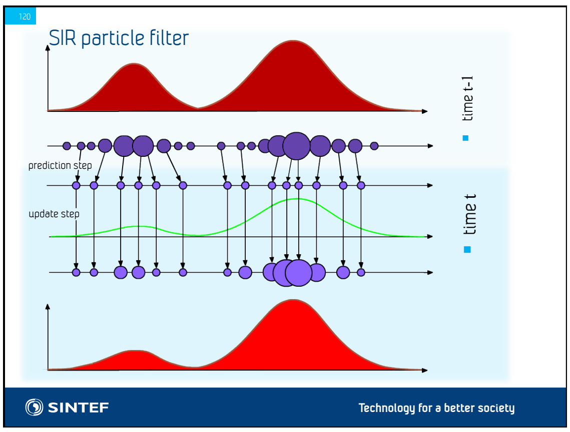

115
Monte Carlo approximation
How can we represent an arbitrary probability density?
$p(x)$
Some complicated PDF we'd like to represent
SINTEF
Technology for a better society


116
Monte Carlo approximation
Represent the PDF non-parametrically, as a set of (weighted) samples!
$p(x)$
Some complicated PDF we'd like to represent
Monte Carlo approximation
$p(x) \approx \sum_{n=1}^{N} w_n \, \delta(x - x_n)$
SINTEF
Technology for a better society


117
Particle approximation
Target distribution
SINTEF
Technology for a better society

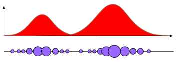

118
Particle approximation
Monte Carlo approximation – **too few samples**
SINTEF
Technology for a better society


119
Particle approximation
Monte Carlo approx – **added samples**
SINTEF
Technology for a better society


120
SIR particle filter
prediction step
update step
time t-1
time t
SINTEF
Technology for a better society

What is a particle?

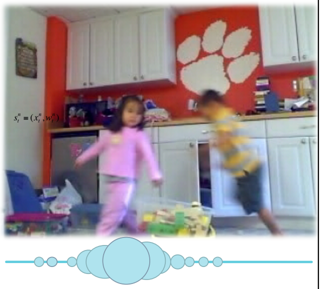
- A "sample" of the posterior



  - **state estimate**
  - **weight**

$s_t^n \equiv (x_t^n, w_t^n)$

- Summing the particles gives an approximation to the target distribution



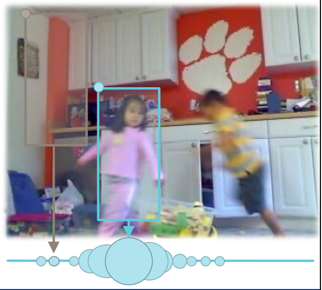
$$p(\mathbf{x}_t \mid Z_t) \approx \sum_{n=1}^{N} w_{t-1}^n \, \delta(\mathbf{x}_t - \mathbf{x}_t^n)$$

What is a particle?

- Each particle contains a
  - **state estimate**
  - **weight**

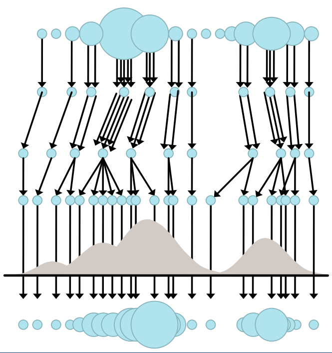
$s_t^n \equiv (x_t^n, w_t^n)$

$$s_t^n \equiv \left(\begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ a \\ h \end{pmatrix}, w_t \right)$$



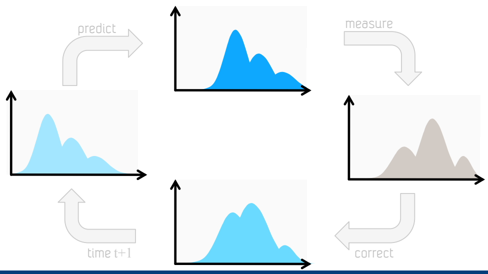
SIR particle filter

- **Begin** with weighted samples from t-1
- **Resample**: draw samples according to $\{w_{t-1}\}^{n=1:N}$
- **Drift**: apply motion model (no noise)
- **Diffuse**: apply noise to spread particles
- **Measure**: weights are assigned by likelihood response
- **Finish**: density estimate



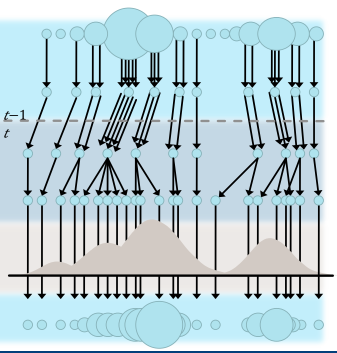
Probability density propagation

- **Notice** similarities to the familiar recursive process



SIR particle filter

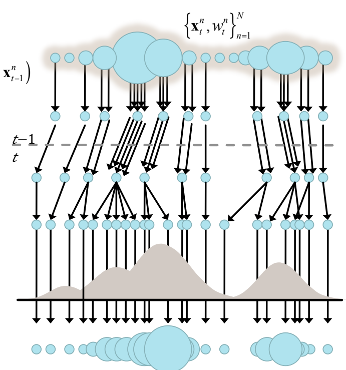
- **Begin** with weighted samples from t-1
- **Resample**: draw samples according to $\{w_{t-1}\}^{n=1:N}$
- **Drift**: apply motion model (no noise)
- **Diffuse**: apply noise to spread particles
- **Measure**: weights are assigned by likelihood response
- **Finish**: density estimate



SIR particle filter

$$p(\mathbf{x}_{t-1} \mid Z_{t-1}) \approx \sum_{n=1}^{N} w_{t-1}^n \, \delta(\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^n)$$

- **Begin** with weighted samples from t-1

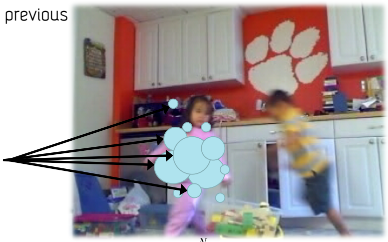


127

Previous estimate

- Receive posterior estimate from previous time step

$\{x_{t-1}^n, w_{t-1}^n\}_{n=1}^N$



$$p(\mathbf{x}_{t-1} \mid Z_{t-1}) \approx \sum_{n=1}^{N} w_{t-1}^n \, \delta(\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^n)$$

SINTEF — Technology for a better society

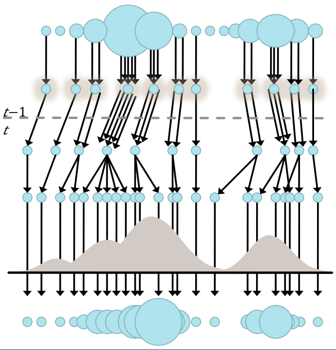
128

SIR particle filter

- **Resample:** draw samples according to $\{w_{t\text{-}1}\}^{n=1:N}$

  N new samples are drawn from the previous set **with replacement**.

  New samples are assigned **uniform weights**.



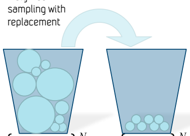
SINTEF — Technology for a better society

129

Resample

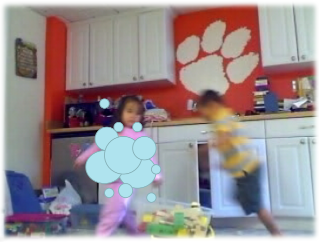
- N new samples are drawn from the previous set **with replacement** to prevent **degeneracy**.
- Repeated samples occur by design.

Weighted sampling with replacement



$\{\mathbf{x}_{t-1}^n, w_{t-1}^n\}_{n=1}^N$ $\{\mathbf{x}_t^n, w_t^n\}_{n=1}^N$

New sample set is given uniform weights



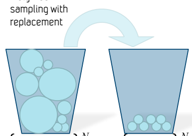
SINTEF — Technology for a better society

130

Resample

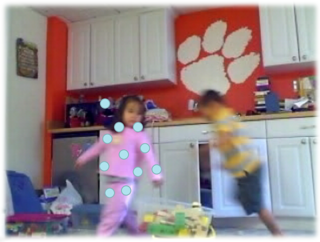
- N new samples are drawn from the previous set **with replacement** to prevent **degeneracy**.
- Repeated samples occur by design.

Weighted sampling with replacement



$\{\mathbf{x}_{t-1}^n, w_{t-1}^n\}_{n=1}^N$ $\{\mathbf{x}_t^n, w_t^n\}_{n=1}^N$

New sample set is given uniform weights

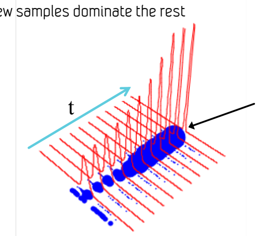


SINTEF — Technology for a better society

131

Degeneracy

- Failing to resample results in **degeneracy**.
  - Iteratively propagating the particles and assigning weights tends to make a few samples dominate the rest



Distribution no longer representative

SINTEF — Technology for a better society

132

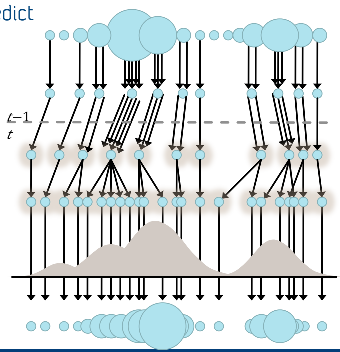
SIR particle filter: predict

- Apply the motion model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ to every particle!

  $$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{w}_t$$

  linear motion model — noise

- **Drift:** apply motion model (no noise)
- **Diffuse:** apply noise to spread particles



SINTEF — Technology for a better society

133

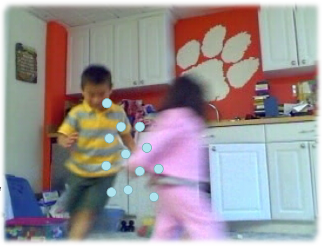
Motion model

- Apply the motion model $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ to every particle!

$$\mathbf{x}_t = \underbrace{\mathbf{F}_t \mathbf{x}_{t-1}}_{\text{linear motion model}} + \underbrace{\mathbf{w}_t}_{\text{noise}}$$

$$\begin{pmatrix} x_t \\ y_t \\ \dot{x}_t \\ \dot{y}_t \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & & \Delta t & \\ & 1 & & \Delta t \\ & & 1 & \\ & & & 1 \end{pmatrix}}_{\mathbf{F}_t} \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ \dot{x}_{t-1} \\ \dot{y}_{t-1} \end{pmatrix} + \mathbf{w}_t$$

$w_t \sim N(0, Q_t)$



SINTEF — Technology for a better society

134

SIR particle filter: measure

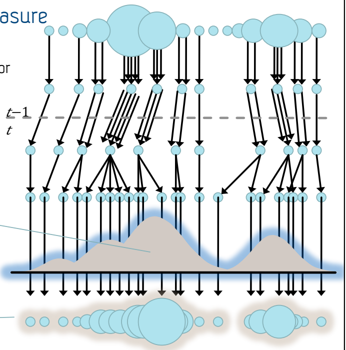
Obtain an observation $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.

Evaluate likelihood that $\mathbf{x}_t$ gave rise to $\mathbf{z}_t$ using observation model.

$p(\mathbf{z}_t | \mathbf{x}_t)$

- **Measure:** weights are proportional to the observation likelihood

$p(\mathbf{x}_t | Z_t)$

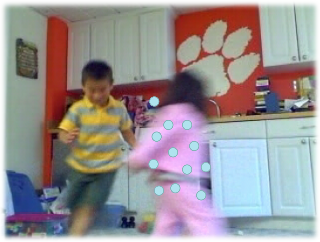


SINTEF — Technology for a better society

135

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.

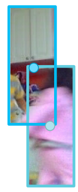


SINTEF — Technology for a better society

136

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.



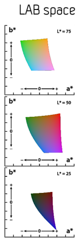
SINTEF — Technology for a better society

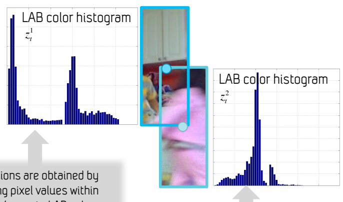
137

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.

LAB space





Observations are obtained by converting pixel values within bounding boxes to LAB colorspace, and concatenating to form an AB channel histogram

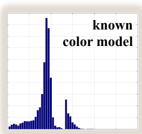
SINTEF — Technology for a better society

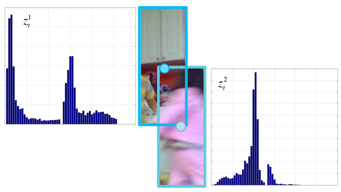
138

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.
- **Evaluate likelihood** that an $\mathbf{x}_t$ gave rise to $\mathbf{z}_t$ using observation model.

$p(z_t | x_t^n) = e^{-\lambda dist(z_t, c)}$





Observation model compares $\mathbf{z}_t$ to a known color model **c** using the **KL divergence**.

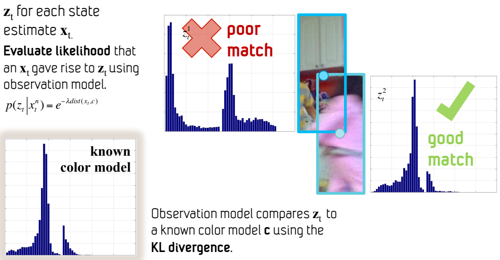
SINTEF — Technology for a better society

139

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.
- **Evaluate likelihood** that an $\mathbf{x}_t$ gave rise to $\mathbf{z}_t$ using observation model.

$p(z_t|x_t^n) = e^{-\lambda dist(x_t,c)}$



Observation model compares $\mathbf{z}_t$ to a known color model **c** using the **KL divergence**.

SINTEF Technology for a better society

140

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.
- **Evaluate likelihood** that an $\mathbf{x}_t$ gave rise to $\mathbf{z}_t$ using observation model.
- **Assign weights** are proportional to the likelihood response

$w_t^n = p(\mathbf{z}_t \mid \mathbf{x}_t^n)$

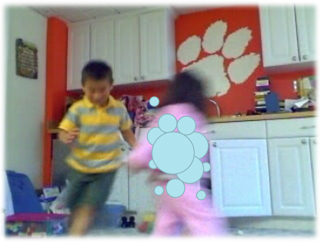
SINTEF Technology for a better society

141

Observation model

- **Obtain observation** $\mathbf{z}_t$ for each state estimate $\mathbf{x}_t$.
- **Evaluate likelihood** that an $\mathbf{x}_t$ gave rise to $\mathbf{z}_t$ using observation model.
- **Assign weights** are proportional to the likelihood response

$w_t^n = p(\mathbf{z}_t \mid \mathbf{x}_t^n)$

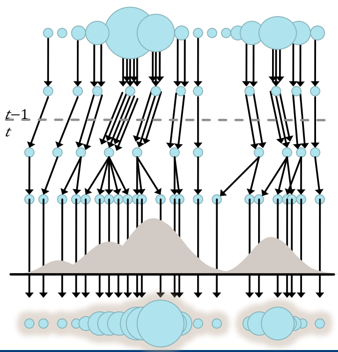


SINTEF Technology for a better society

142

SIR particle filter

- **Begin** with weighted samples from t-1
- **Resample:** draw samples according to $\{w_{t-1}\}^{n=1:N}$
- **Drift:** apply motion model (no noise)
- **Diffuse:** apply noise to spread particles
- **Measure:** weights are assigned by likelihood response
- **Finish:** density estimate

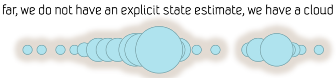


SINTEF Technology for a better society

143

Obtaining a solution

- So far, we do not have an explicit state estimate, we have a cloud of particles!



- How do we extract an answer? It depends…
  - Compute a **mean** or **median** particle
  - Confidence: inverse variance
  - For discrete labels, this does not work!
    - Use the mode?

SINTEF Technology for a better society

144

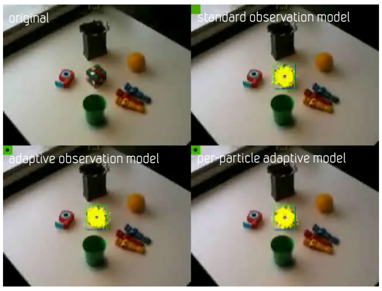
Particle filter

D. Klein, D. Schulz, S. Frintrop, and A. Cremers, Adaptive Real-Time Video Tracking for Arbitrary Objects, International Conference on Intelligent Robots and Systems (IROS), 2010

SINTEF Technology for a better society

145

Particle filter



D. Klein, D. Schulz, S. Frintrop, and A. Cremers, Adaptive Real-Time Video Tracking for Arbitrary Objects ,
International Conference on Intelligent Robots and Systems (IROS), 2010

SINTEF Technology for a better society

146

Summary: particle filters

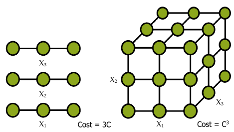
- Represents arbitrary (multi-modal) densities
- Converges to true posterior for nonlinear, non-Gaussian systems
- Efficient: concentrates particles on interesting regions
- Works for many types of state spaces

SINTEF Technology for a better society

147

Summary: particle filters

- Number of samples N is important
  - Use as few as necessary (for efficiency)
  - But use enough to do a good job exploring the state space
- Complexity grows exponentially with dimensionality of the state space



SINTEF Technology for a better society

148

Things to think about…

- Initialization
  - By hand
  - Background subtraction
  - Detection
- Observation models
  - Generative -> render the state on top of the image and compare
  - Discriminative -> classifier or detector score
- Prediction vs Correction
  - If dynamics dominate, cues form the data may be ignored
  - If observation model dominates, tracking is not smooth
- Nonlinear Dynamics
  - Needed for multiple objects, discrete state elements, etc.

SINTEF Technology for a better society

149

Particle filters in action

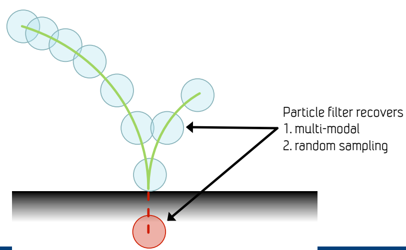
Michael Isard and Andrew Blake CONDENSATION -- conditional density propagation for visual tracking
International Journal of Computer Vision (IJCV), 29, 1, 5--28, (1998)

SINTEF Technology for a better society

150

Particle filters in action

- tracking a ball



SINTEF Technology for a better society

151

Particle filters in action

SINTEF Technology for a better society

Reading material

- Exercise: Try to develop the recursive Bayesian filtering equation from the expressions given in the lecture
- Skim a tutorial that covers Kalman filtering and particle filters
  Arulampalam, M.S.; Maskell, S.; Gordon, N.; Clapp, T.; , *A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, Signal Processing, IEEE Transactions on* , vol. 50, no.2, pp.174-188, Feb 2002, doi: 10.1109/78.978374
  URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=978374&isnumber=21093
- D. Klein, D. Schulz, S. Frintrop, and A. Cremers, *Adaptive Real-Time Video Tracking for Arbitrary Objects* , International Conference on Intelligent Robots and Systems (IROS), 2010

- Credits: This lecture was heavily based on slides from Kevin Smith, ETZH

SINTEF Technology for a better society

Reading materials

- Good survey on tracking algorithms:
  - A. Yilmaz et al. *Object tracking: A survey*, ACM Comput. Surv., 38(4), 2006
- Detect and assign
  - C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(8):747-757, 2000.
  - I.D. Reid and K.R. Connor. Multiview segmentation and tracking of dynamic occluding layers. In BMVC 2005, 2005.
- Mean shift tracker
  - D. Comaniciu, V. Ramesh, and P. Meer, Kernel-based object tracking. *IEEE Trans. Patt. Analy. Mach. Intell. 25*, 564–575, 2003
  - G.R. Bradski et al. *Computer Vision Face Tracking For Use in a Perceptual User Interface, Interface* 2, pp 12–21, 1998 – in OpenCV http://opencv.itseez.com/modules/video/doc/motion_analysis_and_object_tracking.html#camshift
- LK tracker
  - Jianbo Shi and Carlo Tomasi *Good Features to Track*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), 1994, pp. 593 - 600.
  - Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-Backward Error: Automatic Detection of Tracking Failures," *International Conference on Pattern Recognition*, 2010, pp. 23-26.
- Short credit: Some of the slides blatantly "stolen" from Kevin Smith, ETHZ

SINTEF Technology for a better society 153