CENG 4480

Embedded System Development & Applications

Lecture 08: Kalman Filter–2

Bei Yu
CSE Department, CUHK
byu@cse.cuhk.edu.hk

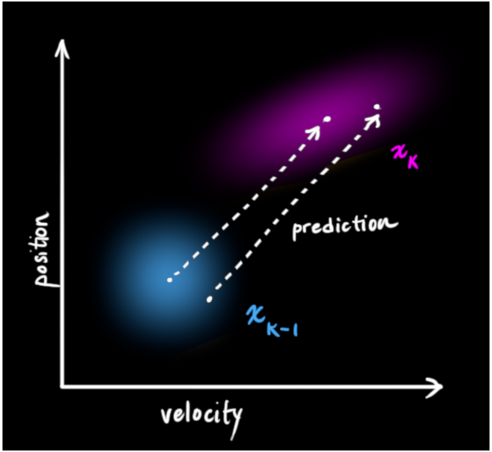
(Latest update: October 27, 2021)

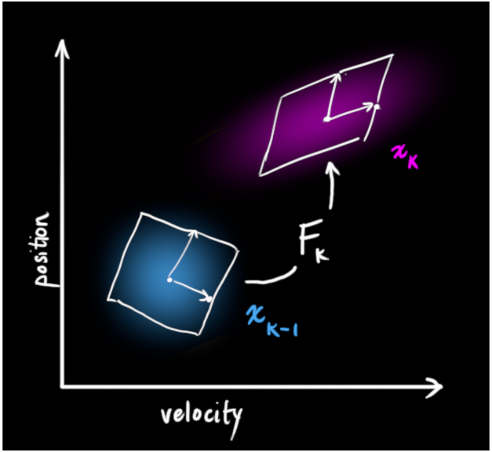
Fall 2021

Kalman Filter

Model with Uncertainty

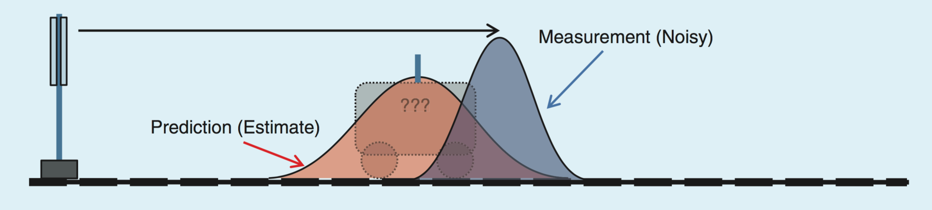
- Model the measurement w. uncertainty (due to noise $\boldsymbol{w}_t$)
- $\boldsymbol{P}_k$: covariance matrix of estimation $\boldsymbol{x}_t$
- On how much we trust our estimated value – the smaller the more we trust



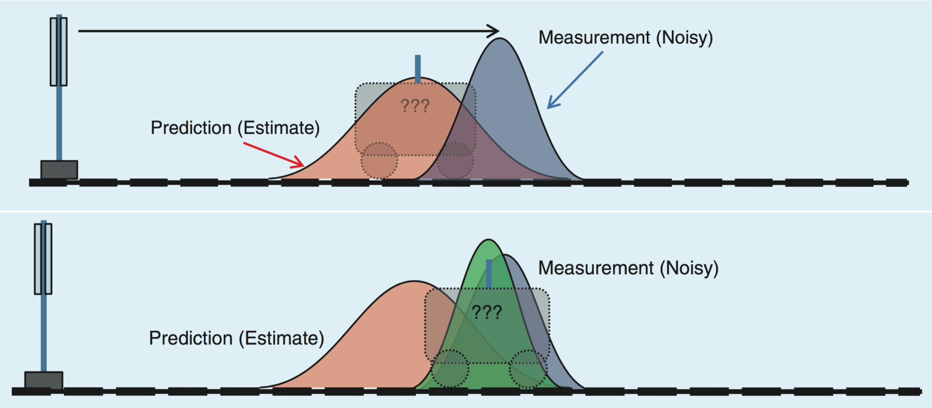


note: here $\boldsymbol{F}_k = \boldsymbol{A}_k$

Fuse Gaussian Distributions



Fuse Gaussian Distributions



Exercise

Given two Gaussian functions $y_1(r;\mu_1,\sigma_1)$ and $y_2(r;\mu_2,\sigma_2)$, prove the product of these two Gaussian functions are still Gaussian.

$$y_1(r;\mu_1,\sigma_1) = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(r-\mu_1)^2}{2\sigma_1^2}} \qquad y_2(r;\mu_2,\sigma_2) = \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(r-\mu_2)^2}{2\sigma_2^2}}$$

Step 1: Prediction

$$\boldsymbol{x}_t^- = \boldsymbol{A}_t \boldsymbol{x}_{t-1} + \boldsymbol{B}_t \boldsymbol{u}_t \tag{1}$$

$$\boldsymbol{P}_t^- = \boldsymbol{A}_t \boldsymbol{P}_{t-1} \boldsymbol{A}_t^\top + \boldsymbol{Q}_t \tag{2}$$

Step 1: Prediction

$$\boldsymbol{x}_t^- = \boldsymbol{A}_t \boldsymbol{x}_{t-1} + \boldsymbol{B}_t \boldsymbol{u}_t \tag{1}$$

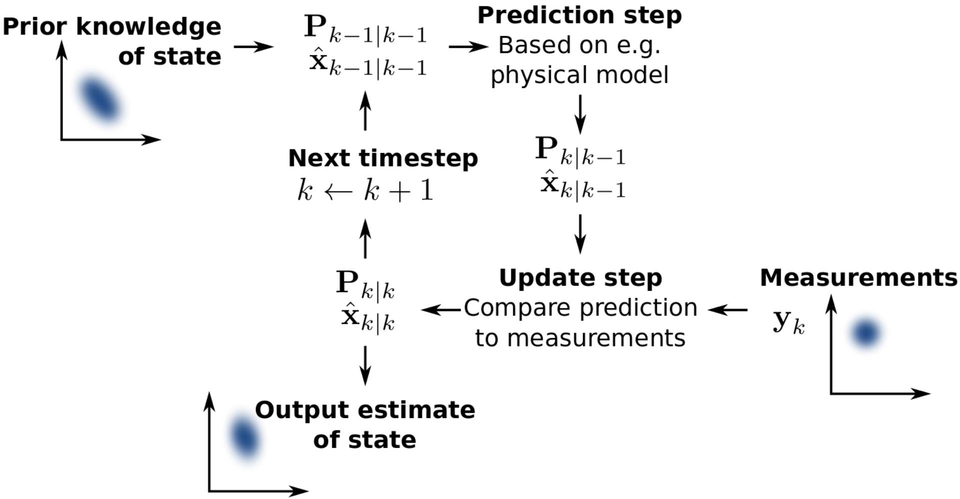
$$\boldsymbol{P}_t^- = \boldsymbol{A}_t \boldsymbol{P}_{t-1} \boldsymbol{A}_t^\top + \boldsymbol{Q}_t \tag{2}$$

Step 2: Measurement Update

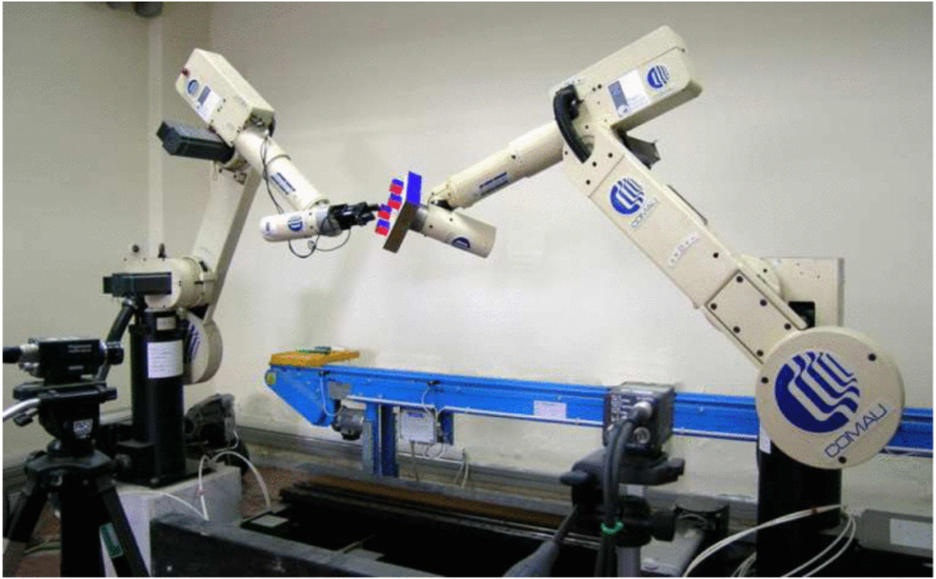
$$\boldsymbol{x}_t = \boldsymbol{x}_t^- + \boldsymbol{K}_t(\boldsymbol{z}_t - \boldsymbol{C}\boldsymbol{x}_t^-) \tag{3}$$

$$\boldsymbol{P}_t = \boldsymbol{P}_t^- - \boldsymbol{K}_t \boldsymbol{C} \boldsymbol{P}_t^- \tag{4}$$

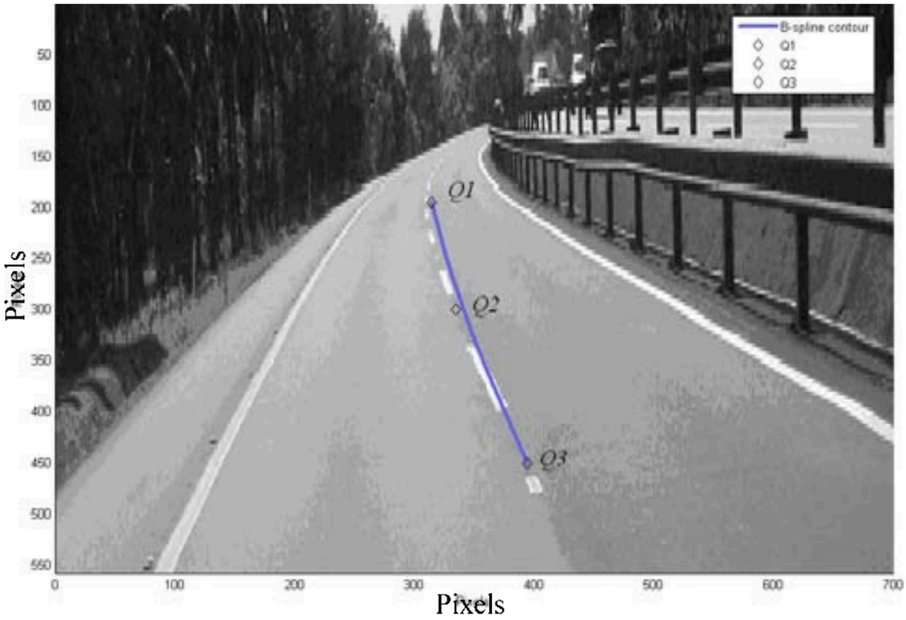
$$\boldsymbol{K}_t = \boldsymbol{P}_t^- \boldsymbol{C}^\top (\boldsymbol{C}\boldsymbol{P}_t^- \boldsymbol{C}^\top + \boldsymbol{R}_t)^{-1} \tag{5}$$


Prior knowledge of state
$\mathbf{P}_{k-1|k-1}$
$\hat{\mathbf{x}}_{k-1|k-1}$
Prediction step
Based on e.g. physical model
$\mathbf{P}_{k|k-1}$
$\hat{\mathbf{x}}_{k|k-1}$
Next timestep
$k \leftarrow k+1$
Update step
Compare prediction to measurements
Measurements
$\mathbf{y}_k$
$\mathbf{P}_{k|k}$
$\hat{\mathbf{x}}_{k|k}$
Output estimate of state

More Applications: Robot Localization



More Applications: Path Tracking



More Applications: Object Tracking

a

The 50th frame

b

The 118th frame

c

The 124th frame

d

The 127th frame

Software

C Implementation

```c
// Kalman filter module
float Q_angle  =  0.001;
float Q_gyro   =  0.003;
float R_angle  =  0.03;

float x_angle = 0;
float x_bias = 0;
float P_00 = 0, P_01 = 0, P_10 = 0, P_11 = 0;
float dt, y, S;
float K_0, K_1;
```

- Q:
- R:
- P:

C Implementation (cont.)

```
float kalmanCalculate(float newAngle, float newRate,int looptime)
{
    dt = float(looptime)/1000;
    x_angle += dt * (newRate - x_bias);
    P_00    += dt * (P_10 + P_01) + Q_angle * dt;
    P_01    += dt * P_11;
    P_10    += dt * P_11;
    P_11    += Q_gyro * dt;

    y   = newAngle - x_angle;
    S   = P_00 + R_angle;
    K_0 = P_00 / S;
    K_1 = P_10 / S;

    x_angle +=  K_0 * y;
    x_bias  +=  K_1 * y;
    P_00 -= K_0 * P_00;
    P_01 -= K_0 * P_01;
    P_10 -= K_1 * P_00;
    P_11 -= K_1 * P_01;

    return x_angle;
}
```