VEHICLE DETECTION AND TRACKING USING KALMAN FILTER

1 – D Model:

1D - ①

My project:

VEHICLE DETECTION
AND
TRACKING

Process:
① Initializes a detector and a tracker.
② Next, the detector localizes the vehicles in each video frame.
③ The tracker is then updated with detection results.
④ Finally the tracker results are annotated and displayed in a video frame.

* Model used:
We use a $\underbrace{\text{pre-trained Mobilenet-coco model}}$.

A single-stage object detection model that goes straight from image pixels to bounding box co-ordinates and class probabilities.

Trained with $\boxed{\text{MS-COCO}}$ dataset: a dataset for image recognition, segmentation and captioning, consisting of > 3 00000 images and 80 obj. classes.

Equations:

Position: $x_K = x_{K-1} + \dot{x}_{K-1}\Delta t + \frac{1}{2}\ddot{x}_{K-1}(\Delta t)^2$

velocity: $\dot{x}_K = \dot{x}_{K-1} + \ddot{x}_{K-1}(\Delta t)$

Since the state vector ~~x_K~~ ~~x_K~~ X_K contains the position and velocity, we can model the example as:

$$X_K = \begin{bmatrix} x_K \\ \dot{x}_K \end{bmatrix} = \begin{bmatrix} x_{K-1} + \dot{x}_{K-1}\Delta t + \frac{1}{2}\ddot{x}_{K-1}(\Delta t)^2 \\ \dot{x}_{K-1} + \ddot{x}_{K-1}\Delta t \end{bmatrix}$$

This linear equation can be written in matrix form as:

$$P_K \; X_K = \begin{bmatrix} x_K \\ \dot{x}_K \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{K-1} \\ \dot{x}_{K-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ (\Delta t) \end{bmatrix} \ddot{x}_{K-1} + W_K$$

This can also be written as:

$$P_K \; X_K = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \overbrace{\overset{P_{K-1}}{X_{K-1}} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ \Delta t \end{bmatrix} \ddot{x}_{K-1}}^{\text{mean}} + W_K \longrightarrow (1)$$

P_K instead of X_K, to avoid confusion.

We already know that,

The Kalman filter is intended to estimate the state of a system at time 'K' using a linear stochastic difference equation assuming that the state of a system at a time 'K' evolved from the prior state at time $K-1$ can be written as

$$\boxed{X_K = A X_{K-1} + B u_{K-1} + W_{K-1}} \longrightarrow (2)$$

noise with covar

A: state matrix

B: control input matrix to the optional control input u_{K-1}

It is always paired with the measurement model Z_K that describes a relationship between the state and measurement at the current step K. It is given as

Observation Equation

$$\boxed{Z_K = H X_K + V_K} \longrightarrow (3)$$

V_K: measurement noise

$y_K = C_K x_K + V_K$

$H \rightarrow$ Transformation matrix that transforms the state into the measurement domain

$$X_{K+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} x_K + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} u_K + W_K$$

* From Eq (1) and (2),

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 1/2(\Delta t^2) \\ \Delta t \end{bmatrix}$$

<u>control</u>: $u_K = a_K$

<u>Noise</u>: W_K

* From (3),

$$Z_K = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_K \\ \dot{x}_K \end{bmatrix} + V_K$$

$$\therefore H = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

* <u>Noise covariance matrix</u>, $\boxed{Q = \begin{bmatrix} \sigma_x^2 & \sigma_x \sigma_{\dot{x}} \\ \sigma_{\dot{x}} \sigma_x & \sigma_{\dot{x}}^2 \end{bmatrix}}$

where,
σ_x = SD of position
$\sigma_{\dot{x}}$ = SD of velocity

* here, we can define the standard - Deviation of the position as the SD of acceleration σ_a multiplied by $\frac{\Delta t^2}{2}$. i.e $\boxed{\sigma_x = \sigma_a \cdot \frac{\Delta t^2}{2}}$

(Since $\frac{\Delta t^2}{2}$ is the effect ~~of~~ that will have on the position)

* $\boxed{\sigma_{\dot{x}} = \sigma_a \cdot \Delta t}$

$$\therefore \boxed{Q = \begin{bmatrix} \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & \Delta t^2 \end{bmatrix} \sigma_a}$$

where, σ_a is the magnitude of SD of the acceleration.

* <u>Measurement of Noise Covariance matrix</u>: (R)

R is a scalar and is equal to the variance of the measurement noise.

$$\therefore \boxed{R = \sigma_z^2}$$

2 – D MODEL:

2D

State matrix (A) and Control matrix (B) :

$$X_K = \begin{bmatrix} x_K \\ y_K \\ \dot{x}_K \\ \dot{y}_K \end{bmatrix} = \begin{bmatrix} x_{K-1} + \dot{x}_{K-1}\Delta t + \frac{1}{2}\ddot{x}_{K-1}\Delta t^2 \\ y_{K-1} + \dot{y}_{K-1}\Delta t + \frac{1}{2}\ddot{y}_{K-1}\Delta t^2 \\ \dot{x}_{K-1} + \ddot{x}_{K-1}\Delta t \\ \dot{y}_{K-1} + \ddot{y}_{K-1}\Delta t \end{bmatrix}$$

$$X_K = \begin{bmatrix} x_K \\ y_K \\ \dot{x}_K \\ \dot{y}_K \end{bmatrix} = \underbrace{\underbrace{\begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{4\times4} \underbrace{\begin{bmatrix} x_{K-1} \\ y_{K-1} \\ \dot{x}_{K-1} \\ \dot{y}_{K-1} \end{bmatrix}}_{4\times1}}_{4\times1} + \underbrace{\underbrace{\begin{bmatrix} \frac{1}{2}(\Delta t)^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix}}_{4\times2} \underbrace{\begin{bmatrix} \ddot{x}_{K-1} \\ \ddot{y}_{K-1} \end{bmatrix}}_{2\times1}}_{4\times1}$$

$$= \begin{bmatrix} x_{K-1} + \Delta t\, \dot{x}_{K-1} \\ y_{K-1} + \Delta t\, \dot{y}_{K-1} \\ \dot{x}_{K-1} \\ \dot{y}_{K-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2\, \ddot{x}_{K-1} \\ \frac{1}{2}\Delta t^2\, \ddot{y}_{K-1} \\ \Delta t\, \ddot{x}_{K-1} \\ \Delta t\, \ddot{y}_{K-1} \end{bmatrix}$$

$$\boxed{\underbrace{X_K}_{\text{current state}} = \underbrace{A}_{\text{Stat matrix}}\ \underbrace{X_{K-1}}_{\text{Previous state}} + \underbrace{B}_{\text{control matrix}}\ \underbrace{a_{K-1}}_{\text{previous acceleration in X and Y direction}}}$$

* <u>Measurement</u> : $\boxed{z_K = H X_K + V_K}$

$$z_K = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_K \\ y_K \\ \dot{x}_K \\ \dot{y}_K \end{bmatrix} + V_K$$

In this case, velocity is not considered in measurement process (we consider only position). But while incorporating Dopplers effect, velocity is considered.
Eg: Airplane and Satellite tracking system

* <u>Transformation matrix</u> :

$$\boxed{H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}$$

* <u>Process noise covariance matrix</u>

$$Q = \begin{array}{cc} \begin{bmatrix} \sigma_x^2 & 0 & \sigma_x \sigma_{\dot{x}} & 0 \\ 0 & \sigma_y^2 & 0 & \sigma_y \sigma_{\dot{y}} \\ \sigma_x \sigma & 0 & \sigma_{\dot{x}}^2 & 0 \\ 0 & \sigma_y \sigma_{\dot{y}} & 0 & \sigma_{\dot{y}}^2 \end{bmatrix} & \begin{array}{c} x \\ y \\ \dot{x} \\ \dot{y} \end{array} \\ \begin{array}{cccc} x & y & \dot{x} & \dot{y} \end{array} & \end{array} = \begin{bmatrix} \frac{\Delta t^2}{4} & 0 & \frac{\Delta t^3}{2} & 0 \\ 0 & \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & 0 & \Delta t^2 & 0 \\ 0 & \frac{\Delta t^3}{2} & 0 & \Delta t^2 \end{bmatrix} \sigma_a^2$$

where, σ_a is magnitude of second derivation of the acceleration that is basically the process noise covariance matrix.

* <u>Measurement noise covariance matrix</u>

In 2D, we suppose that both measurement positions x and y are independent.
$\therefore$ Covariance b/w x and $y = 0$ We look at only the variance in x and y

$$\boxed{R = \begin{array}{cc} \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} & \begin{array}{c} x \\ y \end{array} \\ \begin{array}{cc} x & y \end{array} & \end{array}}$$

PREDICTION AND UPDATION:

(1D)

* Prediction equation : In time update equation we calculate the predicted estimate (a priori state estimate) $\bar{\hat{x}}_K$ and predicted error covariance (a priori error covariance estimate) $\bar{P}_K$

$$\boxed{\bar{\hat{x}}_K = A\hat{x}_{K-1} + Bu_{K-1}}$$

where, $\hat{x}_{K-1}$ is previous estimated state (a posteriori state estimate)

Error covariance matrix :

$$\boxed{\bar{P}_K = AP_{K-1}A^T + \underbrace{Q}_{\text{process noise covariance}}}$$

where, P_{K-1} is the previous estimated error covariance matrix.

* Update equation : We compute Kalman gain as :

$$\boxed{K_K = \bar{P}_K H^T (H\bar{P}_K H^T + R)^{-1}}$$

R is measurement noise covariance

Actual measurement : (z_K)

Measurement residual : $\boxed{(z_K - H\bar{\hat{x}}_K)}$

where, z_K is actual/true measurement

$H\bar{\hat{x}}_K$ is previous estimated measurement

Now, ~~the predicted~~ updated state estimate is,

(Predicted state estimate)

$$\boxed{\hat{x}_K = \underbrace{\bar{\hat{x}}_K}_{\text{priori projected state estimate}} + \overbrace{K_K}^{\text{Kalman gain}} \underbrace{(z_K - H\bar{\hat{x}}_K)}_{\text{Measurement residual}}}$$

updated error covariance (P_K)

$$\boxed{P_K = (I - K_K H)\bar{P}_K}$$

where I is an identity matrix

(x pto y)

* Examining End:

Gaussian is characterized by two parameters: mean(μ) and width of the Gaussian, variance (σ^2)

The quadratic difference b/w query point x ~~and~~ relative to μ is,

$$\boxed{g(x) = c \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}}$$ where, $\boxed{c = \frac{1}{\sqrt{2\pi\sigma^2}}}$ normalizing constant

The task in KF is to maintain μ and σ^2 as best estimate of the location of the object we are trying to find.

* The KF represents all distributions by Gaussians and ~~motion~~ iterations over two different things:
  - MEASUREMENT UPDATES (Corrector equations) ⇒ It involves updating a prior with a product of certain belief.
  - Time update (predictor equations) MOTION UPDATES ⇒ It involves performing a convolution.

* Designing a KF:

t is time step

Initial stage: $\boxed{X_0,\ P_0}$ → Previous state: $\boxed{X_{t-1},\ P_{t-1}}$ → New predicted state:

Matrix (state transition) · Matrix (accelerations) · Errors

$$\boxed{X_t^p = A X_{t-1} + B u_t + W_t}$$

$$\boxed{P_t^p = A P_{t-1} A^T + Q_t}$$

x_0 = state matrix

P_0 = Process cov. matrix

p ⇒ matrix has been updated with a new prediction

↓

Calculate K. Gain(K) and measured Data (Y)

$$\boxed{K = \frac{P_t^p H^T}{H P_t^p H^T + R}}, \quad \boxed{Y_t = C Y_t^m + Z^m}$$

↓

Update process and state matrix

$$\boxed{P_t = (I - KH) P_t^p}$$

$$\boxed{X_t = X_t^p + K[Y - H X_t^p]}$$

← Current state becomes previous state

$$\boxed{X_t = X_{t-1}}$$

$$\boxed{P_t = P_{t-1}}$$

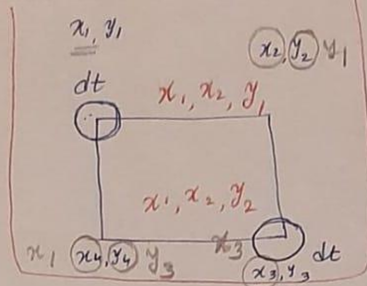
↑ Repeat

VEHICLE DETECTION PROJECT:

EA-1

Eqs:

$$X_K = FX + Bu$$



(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)

1. up – y_1
2. up_dot – $\dot{y}_1$
3. left – x_1
4. left_dot – $\dot{x}_1$
5. down – y_2
6. down_dot – $\dot{y}_2$
7. right – x_2
8. right_dot – $\dot{x}_2$

$$\underbrace{\begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 0 & & 1 \end{bmatrix}}_{8\times 8} \cancel{\begin{bmatrix} x_{1,K-1} \\ y_{1,K-1} \\ x_{3,K-1} \\ y_{3,K-1} \\ \dot{x}_{1,K-1} \\ \dot{y}_{1,K-1} \\ \dot{x}_{3,K-1} \\ \dot{y}_{3,K-1} \end{bmatrix}} \;\&\; \underbrace{\begin{bmatrix} y_{1,K-1} \\ \dot{y}_{1,K-1} \\ x_{1,K-1} \\ \dot{x}_{1,K-1} \\ y_{2,K-1} \\ \dot{y}_{2,K-1} \\ x_{2,K-1} \\ \dot{x}_{2,K-1} \end{bmatrix}}_{8\times 1}$$

(x_1, y_1) ✓

$$\begin{cases} x_{1,K-1} + \dot{x}_{1,K-1}\,\Delta t + \frac{1}{2}\Delta t^2\,\ddot{x}_{1,K-1} \\ y_{1,K-1} + \dot{y}_{1,K-1}\,\Delta t + \frac{1}{2}\ddot{y}_{1,K-1}\,\Delta t^2 \end{cases} \qquad \begin{matrix} \dot{x}_{1,K-1} + \ddot{x}_{1,K-1}\,\Delta t \\ \dot{y}_{1,K-1} + \ddot{y}_{1,K-1}\,\Delta t \end{matrix}$$

(x_2, y_2) ✓

$$\begin{cases} x_{2,K-1} + \dot{x}_{2,K-1}\,\Delta t + \frac{1}{2}\ddot{x}_{2,K-1}\,\Delta t^2 \\ y_{2,K-1} + \dot{y}_{2,K-1}\,\Delta t + \frac{1}{2}\ddot{y}_{2,K-1}\,\Delta t^2 \end{cases} \qquad \begin{matrix} \dot{x}_{2,K-1} + \ddot{x}_{2,K-1}\,\Delta t \\ \dot{y}_{2,K-1} + \ddot{y}_{2,K-1}\,\Delta t \end{matrix}$$

NOT REQUIRED

$\boxed{(x_3, y_3) \,/\, (x_2, y_1)}$

$$\begin{cases} x_{3,K-1} + \ddot{x}_{3,K-1}\,\Delta t + \frac{1}{2}\ddot{x}_{3,K-1}(\Delta t)^2 \\ y_{3,K-1} + \dot{y}_{3,K-1}\,\Delta t + \frac{1}{2}\dot{y}_{3,K-1}\,\Delta t^2 \end{cases} \qquad \begin{matrix} \dot{x}_{3,K-1} + \dot{x}_{3,K-1}\,\Delta t \\ \dot{y}_{3,K-1} + \dot{y}_{3,K-1}\,\Delta t \end{matrix}$$

$\boxed{(x_4, y_4) \,/\, (x_1, y_1)}$

$$\begin{cases} x_{4,K-1} + \dot{x}_{4,K-1}\,\Delta t + \frac{1}{2}\ddot{x}_4(\Delta t^2) \\ y_{4,K-1} + \dot{y}_{4,K-1}\,\Delta t + \frac{1}{2}\ddot{y}_4(\Delta t)^2 \end{cases} \qquad \begin{matrix} \dot{x}_{4,K-1} + \ddot{x}_{4,K-1}\,\Delta t \\ \dot{y}_{4,K-1} + \ddot{y}_{4,K-1}\,\Delta t \end{matrix}$$

Eq-2

Measurement matrix

Eqs $\boxed{X_K = FX + Bu}$

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
y_1 \\ \dot{y}_{1,K-1} \\ x_1 \\ \dot{x}_{1,K-1} \\ y_2 \\ \dot{y}_{2,K-1} \\ x_{2,K-1} \\ \dot{x}_{2,K-1}
\end{bmatrix}
$$

$4 \times 8 = 8 \times 1$

4×1

$$
= \begin{bmatrix}
y_{1,K-1} \\ x_{1,K-1} \\ y_{2,K-1} \\ x_{2,K-1}
\end{bmatrix}
$$

Process state matrix (Equations)

Eq: $\boxed{X_K = FX + Bu}$

$$
\begin{bmatrix}
y_{1,K-1} + \dot{y}_{1,K-1}\Delta t \\
\dot{y}_{1,K-1} \\
x_{1,K-1} + \dot{x}_{1,K-1}\Delta t \\
\dot{x}_{1,K-1} \\
y_{2,K-1} + \dot{y}_{2,K-1}\Delta t \\
\dot{y}_{2,K-1} \\
x_{2,K-1} + \Delta t \cdot \dot{x}_{2,K-1} \\
\dot{x}_{2,K-1}
\end{bmatrix}
\begin{matrix} (1) \\ (2) \\ (3) \\ (4) \\ (5) \\ (6) \\ (7) \\ (8) \end{matrix}
$$