2D Image Processing Exercise Sheet 5 Bayesian Tracking Methods

Group 24

## Task 1. Probability Theory

**1) Three Axioms of Probability Theory:**

$p(A)$ denotes that proposition A is TRUE,

- $0 \leq p(A) \leq 1$
- $p(A = TRUE) = 1 \qquad p(A = FALSE) = 0$
- $p(A \cup B) = p(A) + P(B) + p(A \cap B)$

**2) Joint and Conditional Probability:**

- Joint Probability Distribution

  $p(x, y) = p(X = x \; and \; Y = y)$
- Conditional Probability Distribution

  $p(x|y) = p(X = x \mid Y = y) = \dfrac{p(x,y)}{p(y)}$

  $p(x, y) = p(x|y)\, p(y)$

  If X and Y are independent

  $p(x, y) = p(x)p(y)$

  $p(x|y) = p(x)$

**3) Law of Total Probability:**

The law of total probability is a theorem that, in its discrete case, states if $\{B_n : n=1,2,3\ldots\}$ is a finite or countably infinite partition of a sample space and each event $B_n$ is measurable, then for any event $A$ of the same probability space:

$$p(A) = \sum_n p(A \cap B_n)$$

$$p(A) = \sum_n p(A|B_n) p(B_n)$$

where, $p$ is probability

$A$ is any event

$B_n$ is event

**4) Marginalization of P(x, y) in terms of discrete and continuous random variable:**

| **Discrete Case** | **Continuous Case** |
|---|---|
| $\sum_x p(x) = 1$ | $\int p(x)dx = 1$ |
| $p(x) = \sum_y p(x, y)$ | $p(x) = \int p(x, y)dy$ |
| $p(x) = \sum_y p(x\|y)p(y)$ | $p(x) = \int p(x\|y)p(y)dy$ |

5)

| \ | M | ¬M | Σ |
|---|---|---|---|
| S | 20 | 20 | 40 |

| | | | |
|---|---|---|---|
| ¬S | 35 | 35 | 70 |
| $\Sigma$ | 55 | 55 | |

- $P(M,S) = \frac{20}{55} = \frac{4}{11}$
- $P(M,\neg S) = \frac{35}{55} = \frac{7}{11}$
- $P(M) = \frac{20}{40} = \frac{1}{2}$
- $P(\neg M) = 1 - P(M) = 1 - \frac{20}{40} = \frac{1}{2}$
- $P(S) = \frac{35}{70} = \frac{1}{2}$
- $P(M|S) = \frac{P(M,S)}{P(S)} = \frac{\frac{4}{11}}{\frac{1}{2}} = \frac{8}{11}$

## Task 2: Recursive Bayes Filter and (Robotic) State Estimation:

Task 2:

(1) Recursive Bayes Filter & State Estimation:

W.k.T,

Belief function $\boxed{bel(x_t) = p(x_t : z_{1:t}, u_{1:t})}$

(Sys. state $= x$, observation $= z$, Control $= u$)

Applying Bayesian rule,

$$p(x|y,z) = \frac{p(y|x,z)\,p(x|z)}{p(y|z)}$$

$$\Rightarrow p(x_t : z_{1:t}, u_{1:t}) = \frac{p(z_t|x_t, z_{1:t-1}, u_{1:t})\,p(x_t|z_{1:t-1}, u_{1:t})}{p(z_t|z_{1:t-1}, u_{1:t})} \quad -(1)$$

Applying Markov's rule,

When state is complete,

$$p(z_t|x_{0:t}, z_{1:t-1}, u_{1:t}) = p(z_t|x_t)$$

Markov $\overset{(1)}{\Rightarrow}$ $\eta\, p(z_t|x_t) \cdot p(x_t|z_{1:t-1}, u_{1:t})$

Applying law of total probability,

$$\text{Total probability} = \eta \cdot p(z_t|x_t) \cdot \int p(x_t|x_{t-1}, u_t) \cdot p(x_t|z_{1:t}, u_{1:t})\,dx_{t-1}$$

$$\Rightarrow \boxed{bel(x_t) = \eta \cdot p(z_t|x_t) \cdot \int p(x_t|x_{t-1}, u_t) \cdot bel(x_{t-1}) \cdot dx_{t-1}}$$

(2) where, $bel(x_t)$ = Posterior Conditioning on all available data $(x_t)$

$\eta \cdot p(z_t|x_t)$ = Likelihood | $p(z_t|x_t)$ = Measurement model

$p(x_t|x_{t-1}, u_t)$ = Motion model |

$bel(x_{t-1})$ : Posterior at t-1 step.

## Task 3: Kalman Filter (KF) and Extended Kalman Filter (EKF):

### 1. What are the differences between KF and EKF?

The Kalman filter (KF) is a method based on recursive Bayesian filtering where the noise in your system is assumed Gaussian. The Kalman filter is an efficient recursive filter that is suitable for linear systems.

The Extended Kalman Filter (EKF) is an extension of the classic Kalman Filter for non-linear systems where non-linearity are approximated using the first or second order derivative.

### 2. Explain in a few sentences all of the components of the EKF algorithm.

Output of every iteration cycle:
$\boldsymbol{\mu_t}$ – Calculated State vector
$\boldsymbol{\Sigma_t}$ – Calculated process Covariance matrix

New predicted state and covariance in each iteration cycle:
$\boldsymbol{\mu_t}'$ – Predicted State vector
$\boldsymbol{\Sigma_t}'$ – Predicted Process Covariance matrix

For calculation of new predicted state and covariance during each iteration cycle, the following components are needed:

$\mathbf{H_t}$ – Conversion matrix (from state to measurements)

$\mathbf{Q_t}$ – Process noise covariance matrix (to maintain the process covariance matrix from going too small)

$\mathbf{G_t}$ – Process noise matrix (used in representing non-linear systems)

After the predicted calculations,
$\mathbf{K_t}$ – Kalman gain (calculated from predicted values, $H_t$ and $R_t$ (measurement covariance matrix)) is calculated.

Using the predicted measurements and gain, the output of each iteration cycle is calculated. Terms $\mathbf{H_t}$ and $\mathbf{F_t}$ (state transition matrix) are represented as Jacobian matrix (matrices of partial derivatives) in every iteration cycle and are evaluated using the current predicted states. After this, the matrices can be directly used in kalman filter equations. This process essentially linearizes the non-linear functions to be used in Kalman filters.