

# Week 12

## 1. Implementasi Filter Kalman untuk Estimasi Posisi Robot

**Deskripsi:** Filter Kalman adalah algoritma rekursif yang digunakan untuk memperkirakan keadaan sistem dinamis berdasarkan pengukuran yang mengandung noise. Dalam konteks ini, Filter Kalman diterapkan untuk estimasi posisi robot berdasarkan data pengukuran noisy.

### Hasil Analisis:

- Filter Kalman mampu menghasilkan estimasi posisi yang lebih akurat dibandingkan data pengukuran noisy.
- Matriks gain Kalman membantu menyesuaikan pengaruh pengukuran terhadap estimasi, sehingga noise dapat diminimalkan.
- Visualisasi hasil menunjukkan estimasi yang lebih stabil dan mendekati posisi sebenarnya.

## 2. Implementasi Filter Partikel untuk Estimasi Posisi Robot

**Deskripsi:** Filter partikel adalah metode berbasis Monte Carlo yang digunakan untuk memperkirakan keadaan sistem nonlinear atau non-Gaussian. Dalam tugas ini, filter partikel digunakan untuk estimasi posisi robot dengan menyimulasikan distribusi probabilitas melalui partikel.

### Hasil Analisis:

- Filter partikel dapat menangani noise pengukuran yang kompleks dan distribusi non-linear.
- Resampling memastikan partikel yang relevan mendominasi estimasi, sementara partikel dengan bobot rendah dieliminasi.
- Grafik estimasi menunjukkan bahwa filter partikel memberikan hasil yang akurat meskipun data pengukuran mengandung noise tinggi.

## 3. Implementasi Localization dengan Sensor IMU dan Lidar

**Deskripsi:** IMU (Inertial Measurement Unit) dan Lidar adalah dua sensor utama yang sering digunakan dalam robotika untuk lokalitas. IMU memberikan informasi tentang kecepatan dan orientasi, sedangkan Lidar menyediakan data posisi melalui pengukuran jarak.

### Hasil Analisis:

- Data dari IMU memiliki noise lebih tinggi dibandingkan Lidar, sehingga rentan terhadap drift.
- Penggunaan fusi data, seperti rata-rata berbobot, menghasilkan estimasi posisi yang lebih stabil dan akurat.

- Lokalisasi robot berbasis fusi IMU dan Lidar menunjukkan peningkatan akurasi dibandingkan penggunaan sensor tunggal.

#### **4. Implementasi Simulasi Ekstensi Kalman Filter untuk Navigation**

**Deskripsi:** Ekstensi Kalman Filter (EKF) adalah versi non-linear dari Filter Kalman yang digunakan untuk memperkirakan keadaan sistem dengan dinamika non-linear. EKF diterapkan dalam simulasi untuk navigasi robot.

##### **Hasil Analisis:**

- EKF mampu memperkirakan posisi dan orientasi robot dengan baik meskipun sistem memiliki dinamika non-linear.
- Matriks Jacobian digunakan untuk linearisasi sistem, memungkinkan pembaruan estimasi yang lebih akurat.
- Visualisasi hasil simulasi menunjukkan bahwa EKF dapat mengurangi error posisi selama navigasi.

#### **5. Implementasi Particle Filter untuk Navigation**

**Deskripsi:** Particle filter diterapkan untuk navigasi robot dengan memperkirakan posisi robot berdasarkan distribusi probabilitas partikel. Metode ini sangat cocok untuk kasus dengan noise tinggi atau model non-linear.

##### **Hasil Analisis:**

- Filter partikel untuk navigasi menunjukkan kemampuan untuk memperkirakan posisi secara akurat meskipun data pengukuran memiliki noise tinggi.
- Partikel yang relevan, setelah resampling, memberikan estimasi posisi yang mendekati keadaan sebenarnya.
- Hasil simulasi menunjukkan keunggulan filter partikel dalam menangani skenario navigasi yang kompleks.

## Analisis Kode Webots

### 1. Inisialisasi Robot dan Komponen

- **Robot dan Motor:**
  - Motor kiri dan kanan diatur dalam mode kecepatan dengan posisi tak terbatas (`float('inf')`).
  - Kecepatan awal kedua motor diatur menjadi 0.0.
- **Sensor Jarak:**
  - 8 sensor jarak diaktifkan menggunakan metode `.enable()` dengan langkah waktu tertentu (`TIME_STEP`).
  - Data dari sensor digunakan untuk mendeteksi rintangan di sekitar robot.

### 2. Logika Deteksi dan Penghindaran Rintangan

- Sensor jarak mendeteksi keberadaan rintangan di sisi kiri, kanan, dan depan robot berdasarkan nilai threshold (`OBSTACLE_THRESHOLD`).
- Robot memiliki tiga kondisi utama:
  - **Rintangan di depan:** Robot akan mundur dan berbelok untuk menghindar.
  - **Rintangan di kiri:** Robot akan berbelok ke kanan.
  - **Rintangan di kanan:** Robot akan berbelok ke kiri.
  - **Tidak ada rintangan:** Robot akan berjalan lurus.
- Logika ini memastikan robot dapat bergerak di lingkungan yang dinamis tanpa bertabrakan dengan rintangan.

### 3. Kalman Filter untuk Estimasi Posisi

- **Variabel Awal:**
  - $x$  adalah estimasi posisi awal robot.
  - $P$  adalah ketidakpastian awal estimasi posisi.
- **Proses Kalman Filter:**
  - **Prediksi:** Robot memperkirakan posisi berdasarkan keadaan sebelumnya ( $x_{pred}$ ) dan noise proses.
  - **Update:** Posisi diperbarui berdasarkan pengukuran sensor ( $z$ ) dan gain Kalman ( $K$ ).
  - Gain Kalman menentukan seberapa besar pengukuran sensor memengaruhi estimasi posisi.

- **Pengukuran Sensor:**

- Nilai pengukuran ( $z$ ) adalah nilai terkecil dari sensor jarak, yang merepresentasikan jarak terdekat dengan rintangan.

#### 4. Output

- Estimasi posisi ( $x$ ) dan nilai sensor dicetak di setiap iterasi loop. Hal ini memberikan gambaran mengenai dinamika posisi robot dan interaksi dengan lingkungannya.

### Hasil Analisis

#### 1. Keunggulan Kalman Filter:

- Kalman Filter menghasilkan estimasi posisi robot yang lebih akurat dengan menggabungkan pengukuran sensor (yang mengandung noise) dengan model pergerakan robot.
- Noise proses dan pengukuran diminimalkan secara dinamis melalui mekanisme pembaruan gain Kalman.

#### 2. Efisiensi Deteksi Rintangan:

- Logika berbasis sensor jarak memungkinkan robot untuk secara adaptif menghindari rintangan di lingkungannya.
- Kombinasi Kalman Filter dan penginderaan jarak membuat robot lebih responsif terhadap perubahan lingkungan.

#### 3. Peningkatan yang Disarankan:

- **Model Pergerakan:** Saat ini, estimasi pergerakan robot ( $u$ ) disetel ke nol. Dengan memasukkan data aktual pergerakan, seperti perubahan kecepatan atau orientasi, estimasi posisi bisa lebih akurat.
- **Threshold Sensor:** Threshold untuk deteksi rintangan (OBSTACLE\_THRESHOLD) dapat disesuaikan berdasarkan lingkungan simulasi.
- **Pemanfaatan Data Sensor:** Data dari beberapa sensor dapat digabungkan (fusi data) untuk meningkatkan keakuratan estimasi posisi.