

# ANALISIS

## 1. camera.c

### Deskripsi:

Program ini mengontrol robot untuk mendeteksi **blob warna tertentu (merah, hijau, biru)** di area pandangan kamera. Robot mendeteksi warna dominan di tengah-tengah pandangan kamera dan menggerakkan roda untuk berbelok atau berhenti tergantung keberadaan blob tersebut.

### Penjelasan Kode:

- Kamera diaktifkan dengan `wb_camera_enable(camera, time_step)` untuk menangkap gambar.
- **Deteksi Blob:**
  - Program memindai piksel kamera pada area tertentu, menjumlahkan intensitas warna merah, hijau, dan biru.
  - Jika salah satu warna lebih dominan dari yang lain, maka blob warna itu terdeteksi.
- Robot berhenti ketika blob ditemukan (`pause_counter` digunakan untuk mencegah pendeteksian berulang).
- Warna yang terdeteksi disimpan dalam gambar menggunakan `wb_camera_save_image()`.

### Tujuan Simulasi:

Deteksi warna di lingkungan robot dengan kamera, lalu lakukan tindakan sesuai warna yang ditemukan.

---

## 2. camera\_auto\_focus.c

### Deskripsi:

Program ini menggunakan fitur **auto-focus kamera** untuk mengatur fokus berdasarkan jarak objek yang terdeteksi dengan sensor jarak.

### Penjelasan Kode:

- Kamera diaktifkan, dan fokusnya diatur menggunakan `wb_camera_set_focal_distance()`.
- **Sensor Jarak:**
  - Robot menggunakan sensor jarak (`wb_distance_sensor_get_value()`), mengukur jarak objek di depan kamera.
  - Fokus kamera diatur sesuai dengan nilai jarak tersebut.

### Tujuan Simulasi:

Menunjukkan bagaimana kamera dapat digunakan bersama sensor jarak untuk memperbaiki gambar dengan fokus otomatis.

---

### 3. camera\_motion\_blur.c

#### Deskripsi:

Robot memanfaatkan kamera untuk mendeteksi blob warna sambil bergerak, sehingga simulasi menghasilkan **motion blur** pada gambar kamera.

#### Penjelasan Kode:

- Mirip dengan *camera.c*, tetapi dengan fokus pada visualisasi pergerakan kamera.
- Warna yang terdeteksi disimpan, dan program memanfaatkan *blur effect* yang melekat pada pergerakan.

#### Tujuan Simulasi:

Menunjukkan efek blur saat kamera menangkap gambar dalam kondisi robot bergerak.

---

### 4. camera\_noise\_mask.c

#### Deskripsi:

Simulasi ini memperkenalkan **masking noise** pada gambar kamera, yang bertujuan untuk menunjukkan efek gangguan visual dan bagaimana robot dapat tetap mendeteksi warna.

#### Penjelasan Kode:

- Deteksi blob menggunakan logika serupa pada *camera.c*.
- Namun, tambahan noise memengaruhi akurasi deteksi blob warna.

#### Tujuan Simulasi:

Mengajarkan bagaimana robot beradaptasi dalam lingkungan dengan gangguan visual.

---

### 5. camera\_recognition.c

#### Deskripsi:

Program ini menggunakan fitur **camera recognition** untuk mengenali objek di lingkungan simulasi berdasarkan atribut seperti model, ukuran, dan posisi.

#### Penjelasan Kode:

- **Recognition Mode:** Diaktifkan dengan `wb_camera_recognition_enable(camera)`.
- Informasi objek yang dikenali diperoleh dari `wb_camera_recognition_get_objects()` dan dicetak, termasuk:
  - **Model:** Nama objek.
  - **ID:** Identitas unik objek.
  - **Posisi:** Posisi relatif dalam koordinat kamera.
  - **Orientasi:** Rotasi objek relatif terhadap robot.

- **Ukuran:** Dimensi objek di dunia nyata dan gambar kamera.

#### **Tujuan Simulasi:**

Penggunaan kamera untuk mengenali objek dan mengidentifikasi atribut mereka secara rinci.

---

### **6. camera\_segmentation.c**

#### **Deskripsi:**

Program ini menggunakan **segmented image** untuk membedakan objek yang dikenali dan menampilkan gambar hasil segmentasi pada layar robot.

#### **Penjelasan Kode:**

- Segmentation diaktifkan dengan `wb_camera_recognition_enable_segmentation(camera)`.
- Gambar segmentasi diperoleh dengan `wb_camera_recognition_get_segmentation_image()` dan ditampilkan dengan `wb_display_image_paste()`.
- Robot terus bergerak sambil menunjukkan hasil segmentasi dalam layar simulasi.

#### **Tujuan Simulasi:**

Demonstrasi kemampuan segmentasi visual untuk memisahkan objek di lingkungan simulasi.

---

### **7. spherical\_camera.c**

#### **Deskripsi:**

Simulasi ini melibatkan kamera berbentuk **spherical** (kamera 360 derajat) untuk mendeteksi blob warna dan arah mereka dalam ruang 3D.

#### **Penjelasan Kode:**

- Kamera mendeteksi blob warna merah, hijau, dan biru di area pandangan 360 derajat.
- Posisi blob dihitung berdasarkan koordinat piksel dan dikonversi menjadi sudut menggunakan `atan()`.

#### **Tujuan Simulasi:**

Mengeksplorasi aplikasi kamera spherical dalam simulasi navigasi robot.

---

### **Kesimpulan Analisis**

Simulasi ini menunjukkan penggunaan kamera robot dalam berbagai aplikasi, meliputi:

1. **Blob Detection:** Deteksi warna spesifik.
2. **Auto-focus:** Penyesuaian kamera berdasarkan jarak objek.
3. **Motion Blur:** Simulasi pergerakan.
4. **Noise Masking:** Penanganan gangguan visual.

5. **Object Recognition:** Identifikasi objek di lingkungan.
6. **Image Segmentation:** Pemisahan objek dari latar belakang.
7. **360-Degree Vision:** Deteksi arah objek dalam ruang.