

Tugas 9

Nama : muhammad makhlufi makbullah

Kelas : TK 45 01

NIM : 1103210171

1. Implementasi Segmentasi Kamera pada Robot Menggunakan Webots

Segmentasi Kamera adalah proses memisahkan area tertentu dalam gambar berdasarkan karakteristik seperti warna, intensitas, atau tekstur. Pada Webots, ini dapat dilakukan dengan menggunakan kamera untuk menangkap gambar, kemudian memproses gambar tersebut (misalnya, dengan deteksi warna untuk membedakan objek).

Penjelasan Implementasi:

1. Tujuan: Membagi gambar yang diambil kamera menjadi bagian-bagian berdasarkan warna.
2. Proses:
 - Gunakan kamera bawaan robot untuk mengambil gambar.
 - Terapkan filter berbasis warna (misalnya, deteksi objek berwarna merah).
 - Tampilkan hasil segmentasi di jendela konsol atau GUI.

Kode untuk Segmentasi Kamera di Webots:

```
from controller import Robot, Camera, Display  
  
import numpy as np
```

```
# Inisialisasi robot dan perangkat
```

```
robot = Robot()
```

```
time_step = int(robot.getBasicTimeStep())
```

```
# Kamera
```

```
camera = robot.getDevice("camera")
```

```
camera.enable(time_step)
```

```

# Display untuk menampilkan segmentasi
display = robot.getDevice("display")
while robot.step(time_step) != -1:
    # Ambil gambar dari kamera
    image = camera.getImage()
    width = camera.getWidth()
    height = camera.getHeight()

    # Konversi ke array numpy untuk manipulasi
    img_array = np.frombuffer(image, dtype=np.uint8).reshape((height, width, 4))

    # Deteksi warna merah (contoh segmentasi warna)
    red_channel = img_array[:, :, 0]
    green_channel = img_array[:, :, 1]
    blue_channel = img_array[:, :, 2]
    mask_red = (red_channel > 150) & (green_channel < 100) & (blue_channel < 100)

    # Buat gambar biner berdasarkan mask
    segmented_img = np.zeros_like(img_array)
    segmented_img[mask_red] = [255, 0, 0, 255] # Warna merah dengan alpha 255

    # Tampilkan hasil segmentasi di display
    display.imageNew(segmented_img.tobytes(), Display.RGBA, width, height)
    display.imagePaste(display.imageLoad(segmented_img.tobytes()), 0, 0)

```

Penjelasan Kode:

- Kamera menangkap gambar dan dikonversi ke format array dengan **numpy**.
- Deteksi warna merah dilakukan dengan memeriksa nilai **RGB**.
- Gambar biner hasil segmentasi ditampilkan di perangkat **Display**.

2. Implementasi Penggunaan Kamera untuk Deteksi Bola pada Robot Menggunakan Webots

Deteksi Bola adalah kasus spesifik dari segmentasi, di mana kita mencari bentuk bola berdasarkan warna dan geometri.

Penjelasan Implementasi:

1. **Tujuan:** Mendeteksi bola berdasarkan warnanya (misalnya, bola merah) dan lokasi dalam gambar.
2. **Proses:**
 - Gunakan kamera untuk mengambil gambar.
 - Deteksi warna bola untuk membedakan dari lingkungan.
 - Gunakan algoritma sederhana (seperti deteksi lingkaran) untuk mengonfirmasi bentuk bola.

Kode untuk Deteksi Bola di Webots:

```
from controller import Robot, Camera
import numpy as np
import cv2 # OpenCV untuk deteksi lingkaran
```

```
# Inisialisasi robot dan perangkat
```

```
robot = Robot()
```

```
time_step = int(robot.getBasicTimeStep())
```

```
# Kamera
```

```
camera = robot.getDevice("camera")
```

```
camera.enable(time_step)
```

```
while robot.step(time_step) != -1:
```

```
    # Ambil gambar dari kamera
```

```
    image = camera.getImage()
```

```
    width = camera.getWidth()
```

```
    height = camera.getHeight()
```

```

# Konversi ke array numpy
img_array = np.frombuffer(image, dtype=np.uint8).reshape((height, width, 4))

# Konversi ke format BGR untuk OpenCV
img_bgr = img_array[:, :, :3][:, :, ::-1] # RGB -> BGR

# Konversi ke HSV untuk deteksi warna
hsv = cv2.cvtColor(img_bgr, cv2.COLOR_BGR2HSV)

# Mask untuk warna merah
lower_red = np.array([0, 100, 100])
upper_red = np.array([10, 255, 255])
mask_red = cv2.inRange(hsv, lower_red, upper_red)

# Deteksi lingkaran
circles = cv2.HoughCircles(mask_red, cv2.HOUGH_GRADIENT, dp=1.2, minDist=30,
param1=50, param2=30, minRadius=10, maxRadius=50)

if circles is not None:
    circles = np.uint16(np.around(circles))
    for circle in circles[0, :]:
        x, y, r = circle
        print(f"Bola terdeteksi di koordinat: ({x}, {y}) dengan radius: {r}")

```

Penjelasan Kode:

- Kamera menangkap gambar dan memprosesnya menjadi format OpenCV.
- Konversi gambar ke ruang warna HSV untuk mendeteksi warna merah.
- Gunakan metode Hough Transform untuk mendeteksi lingkaran.
- Koordinat dan radius bola dicetak di konsol.