

Laporan Tugas 9

1. Segmentasi Kamera di Robot Menggunakan Webots

1.1 Tujuan

Melakukan segmentasi gambar yang diambil kamera robot berdasarkan warna, misalnya mendeteksi objek berwarna merah.

1.2 Proses

1. Kamera robot menangkap gambar.
2. Gambar dikonversi ke array numpy untuk memudahkan manipulasi.
3. Pendeteksian warna merah dilakukan dengan memeriksa nilai RGB.
4. Hasil segmentasi warna ditampilkan pada display di GUI Webots.

1.3 Kode Implementasi Segmentasi Kamera

python

Copy code

```
from controller import Robot, Camera, Display
```

```
import numpy as np
```

```
robot = Robot()
```

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

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

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

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

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

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

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

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

```

img_array = np.frombuffer(image, dtype=np.uint8).reshape((height, width, 4))

# Deteksi warna merah
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)

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

display.imageNew(segmented_img.tobytes(), Display.RGBA, width, height)
display.imagePaste(display.imageLoad(segmented_img.tobytes()), 0, 0)

```

1.4 Penjelasan Kode

- Gambar yang diambil kamera dikonversi ke array numpy.
- Segmentasi dilakukan berdasarkan deteksi warna merah.
- Hasil ditampilkan di jendela display robot.

2. Deteksi Bola di Robot Menggunakan Webots

2.1 Tujuan

Mendeteksi bola berdasarkan warna dan bentuknya, misalnya bola berwarna merah.

2.2 Proses

1. Kamera robot menangkap gambar.
2. Gambar dikonversi ke format yang didukung oleh OpenCV.
3. Masking warna merah dilakukan di ruang warna HSV.
4. Algoritma Hough Circle Transform digunakan untuk mendeteksi lingkaran yang menunjukkan keberadaan bola.

2.3 Kode Implementasi Deteksi Bola

python

Copy code

```
from controller import Robot, Camera
```

```
import numpy as np
```

```
import cv2
```

```
robot = Robot()
```

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

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

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

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

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

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

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

```
    img_array = np.frombuffer(image, dtype=np.uint8).reshape((height, width, 4))
```

```
    # Konversi ke format BGR untuk OpenCV
```

```
    img_bgr = img_array[:, :, :3][:, :, ::-1]
```

```
    # Konversi ke HSV untuk deteksi warna
```

```
    hsv = cv2.cvtColor(img_bgr, cv2.COLOR_BGR2HSV)
```

```
    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}")
```

2.4 Penjelasan Kode

- Gambar dikonversi ke ruang warna HSV untuk deteksi warna.
- Lingkaran pada gambar dideteksi menggunakan metode Hough Transform.
- Koordinat dan radius bola dicetak di konsol jika bola terdeteksi.