

Nama : A. Muh. Faizhul Islam

Kelas : TK-45-G09

NIM : 1103210074

Berikut adalah tugas-tugas yang dikerjakan sesuai dengan soal:

1. Camera Robot untuk Mendeteksi Blob Warna (Merah, Hijau, dan Biru)

Kamera robot digunakan untuk memisahkan warna merah, hijau, dan biru pada gambar yang diambil.

Implementasi dilakukan dengan membaca data warna menggunakan filter RGB pada array gambar.

2. Camera Robot dengan Fokus Kamera Berdasarkan Objek yang Ada di Depannya

- Robot mendeteksi objek terdekat dan mengatur fokus kamera untuk mendapatkan detail objek tersebut.
- Proses ini memanfaatkan jarak objek dari sensor atau algoritma pengenalan objek.

3. Camera Robot Deteksi Blob Berwarna pada Robot dengan Efek Motion Blur Kamera

- Mengimplementasikan segmentasi warna untuk mendeteksi blob berwarna pada gambar dengan efek motion blur.
- Menggunakan teknik preprocessing seperti pengurangan noise sebelum segmentasi.

4. Robot dengan Kamera: Deteksi Blob Berwarna dengan Noise Mask

- Memanfaatkan noise mask untuk meningkatkan akurasi deteksi blob berwarna.
- Noise mask digunakan untuk mengabaikan area gambar yang tidak relevan.

5. Deteksi Objek dengan Kamera dan Pengenalan Objek pada Robot

- Objek dikenali berdasarkan fitur tertentu seperti bentuk, ukuran, atau warna.
- Implementasi melibatkan pemrosesan lanjutan menggunakan algoritma pengenalan objek seperti template matching.

6. Implementasi Segmentasi Kamera pada Robot Menggunakan Webots

- Robot menggunakan kamera untuk membagi gambar menjadi area-area berbeda berdasarkan

karakteristik visual.

- Misalnya, membedakan area jalan, dinding, dan rintangan.

7. Implementasi Penggunaan Kamera Bola pada Robot Menggunakan Webots

- Kamera mendeteksi bola berdasarkan warna dan bentuk lingkaran.
- Proses melibatkan algoritma segmentasi warna dan deteksi lingkaran menggunakan transformasi Hough.

Kode Python untuk implementasi tugas-tugas ini dijelaskan di bawah.

Kode Segmentasi Kamera

```

# Kode Segmentasi Kamera

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("ca
mera")
camera.enable(time_
step)

# Display untuk menampilkan
segmentasi display =
robot.getDevice("display")

while
robot.step(time_s
tep) != -1: #
Ambil gambar

```

dari kamera

image =

camera.getImage

() width =

camera.getWidth

() height =

camera.getHeigh

t()

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)
```

```
# Kode Deteksi Bola 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.getBasicTimeSte  
p())
```

```
# Kamera camera =  
robot.getDevice("ca  
mera")  
camera.enable(time_  
step)
```

```
while  
robot.step(time_s  
tep) != -1: #  
Ambil gambar
```

dari kamera

image =

camera.getImage

() width =

camera.getWidth

() height =

camera.getHeigh

t()

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(circl
```

```
es)) for circle in circles[0,
```

```
:]
```

```
    x, y, r = circle print(f"Bola terdeteksi di
```

```
koordinat: ({x}, {y}) dengan radius: {r}")
```