

Nama : Irman Prayista

NIM : 1103210094

Kelas : TK-45-02

Hasil Analisis Simulasi

1. Simulasi *Information Extraction* Menggunakan *Python* dan *OpenCV*

Pada simulasi *Information Extraction* menggunakan *Python* dan *OpenCV*, terdapat simulasi ekstraksi garis dengan *Hough Transform* yang berfungsi untuk menampilkan dan mendeteksi garis menggunakan *Hough Transform*. Terdapat fungsi *hough_line_detection()* yang berfungsi untuk menerapkan *Hough Transform* pada gambar yang di unggah. Selain itu, program ini berfungsi dalam menangani kesalahan dan mendeteksi garis agar lebih *robust* dalam menghadapi berbagai jenis gambar.

Terdapat juga simulasi *Template Matching* untuk mendeteksi objek. Pada simulasi ini terdapat fungsi *template_matching()* yang digunakan untuk melakukan pencocokan *template* dengan gambar asli dan menggambar kotak hijau pada *template* di sekitar are gambar. Program ini mengasumsikan *template* cocok dengan objek dalam gambar dengan tingkat kecocokan 0.8 atau lebih.

Terdapat juga simulasi lainnya seperti Pembuatan Pyramid Gambar, program ini akan mengunggah gambar dan membuat Pyramid gambar untuk menunjukkan berbagai resolusi. Terdapat fungsi *image_pyramid()* yang akan membuat beberapa tingkatan resolusi gambar menggunakan teknik *pyrDown* untuk menurunkan resolusi dan *pyrUp*. Lalu gambar akan ditampilkan dalam bentuk sub plot.

Terdapat juga simulasi lainnya seperti deteksi lingkaran menggunakan *Hough Transform*. Pada program ini akan mengunggah gambar asli dan terdapat fungsi *hough_circle_detection()* yang akan memproses gambar dalam bentuk format *grayscale* dan menerapkan filter median untuk mengurangi *noise*. Setelah itu, fungsi *cv2.HoughCircles()* digunakan untuk mendeteksi lingkaran dalam gambar. Hasilnya adalah gambar dengan lingkaran yang terdeteksi ditandai dengan warna hijau dan titik lingkaran ditandai dengan warna merah.

Selain itu, terdapat simulasi ekstraksi warna dominan pada gambar. Program ini akan mengekstrak warna dominan pada gambar asli yang diunggah menggunakan metode *K-Means Clustering*. Terdapat fungsi *extract_dominant_color()* yang berfungsi mengubah gambar menjadi format RGB dan diubah menjadi bentuk vektor untuk analisis warna. Lalu metode *K-*

Means digunakan untuk mengelompokkan warna gambar menjadi beberapa kluster. Hasil gambar akan ditampilkan dalam bentuk *rectangle*.

Terdapat juga simulasi deteksi kontur pada gambar, program ini terdapat fungsi *contour_detection()* yang digunakan mengubah gambar menjadi *grayscale* lalu menggunakan *thresholding* untuk mendeteksi area yang kontras. Kemudian, kontur – kontur yang terdeteksi pada gambar asli akan diwarnai hijau.

2. Simulasi *Information Extraction* Menggunakan *Webots*

Simulasi ini digunakan untuk implementasi kontrol robot menggunakan *Webots*, dengan integrasi sensor LIDAR dan ultrasonik untuk navigasi dan penghindaran hambatan. Robot diinisialisasi dengan dua motor roda (kiri dan kanan), yang kecepatannya diatur berdasarkan data dari sensor jarak ultrasonik. Sensor LIDAR digunakan untuk memperoleh data jarak objek di sekitar. Koefisien dalam matriks *coefficients* digunakan untuk menghitung kecepatan motor berdasarkan pembacaan sensor jarak, sehingga memungkinkan perilaku robot yang responsif terhadap lingkungan.