

1. Tujuan Tugas

Tugas utama adalah untuk mengembangkan sebuah sistem pemindaian dokumen di **Webots** menggunakan teknik **image processing**. Sistem ini bertujuan untuk mengambil gambar dokumen dari **kamera robot** yang bergerak di atas **conveyor belt**, melakukan **segmentasi** dan **transformasi perspektif** untuk menghasilkan tampilan **top-down** yang jelas dan benar dari dokumen yang dipindai.

Tugas ini terdiri dari beberapa bagian utama:

1. **Mengambil gambar dari kamera robot.**
2. **Melakukan segmentasi warna** untuk mendeteksi objek dokumen.
3. **Transformasi perspektif** untuk mendapatkan tampilan **top-down** dari dokumen.
4. **Menampilkan hasilnya di layar Webots.**
5. **Menyimpan gambar hasil pemrosesan** ke disk untuk analisis lebih lanjut.

2. Proses Implementasi

2.1 Pengaturan Webots

Sebagai simulator, **Webots** digunakan untuk mensimulasikan dunia nyata, di mana robot memiliki **kamera** dan **display** untuk melakukan pemindaian dokumen. **File dunia (.wbt)** dikonfigurasi untuk memiliki robot dengan kamera dan display yang aktif.

- **Kamera:** Mengambil gambar dari dunia simulasi dan digunakan sebagai input untuk pemrosesan citra.
- **Display:** Menampilkan hasil pemrosesan gambar, termasuk dokumen yang dipindai setelah perspektifnya dikoreksi.

2.2 Pengambilan Gambar dan Pemrosesan Citra

Bagian penting dalam tugas ini adalah **pemrosesan citra** yang dilakukan untuk mendapatkan gambar dokumen yang dapat dipindai dalam **top-down view**.

- **Segmentasi Warna:** Menggunakan fungsi **segment_by_color()**, gambar yang diambil dari kamera diproses dengan teknik **segmentation** berdasarkan warna menggunakan ruang warna **HSV**. Tujuan dari segmentasi ini adalah untuk menyorot objek dokumen berdasarkan rentang warna tertentu yang telah ditentukan.
 - **Masalah yang Dihadapi:** Rentang warna yang digunakan dalam segmentasi mungkin perlu disesuaikan, tergantung pada warna dokumen yang ingin dipindai. Jika rentang warna terlalu sempit atau terlalu luas, deteksi mungkin gagal.
- **Mendeteksi dan Mengoreksi Perspektif:** Fungsi **get_warped_document()** digunakan untuk menemukan kontur objek dalam gambar dan menghitung transformasi perspektif. Jika dokumen terdeteksi dengan benar, transformasi akan diterapkan untuk meratakan gambar dokumen menjadi tampilan **top-down** yang

sesuai.

- **Masalah yang Dihadapi:** Deteksi kontur dan perhitungan perspektif sering kali sensitif terhadap kualitas gambar. Jika gambar yang diambil dari kamera buram atau tidak cukup jelas, deteksi kontur dapat gagal. Oleh karena itu, pencahayaan dan kualitas gambar penting untuk mencapai hasil yang baik.

2.3 Menampilkan Hasil di Webots

Setelah gambar diproses dan perspektif dikoreksi, hasilnya ditampilkan di layar **display Webots**. Gambar yang telah diproses dan dipindai akan ditampilkan di layar robot sehingga pengguna dapat melihat tampilan **top-down** dari dokumen.

- **Display:** Hasil pemrosesan gambar ditampilkan di display robot menggunakan fungsi `display_numpy_image()` yang mengonversi gambar yang diproses menjadi format yang sesuai untuk ditampilkan di Webots.

2.4 Menyimpan Gambar ke Disk

Untuk analisis lebih lanjut atau penyimpanan hasil pemindaian, gambar yang diproses dapat disimpan ke disk menggunakan fungsi `save_image()`. Gambar disimpan di folder `C:\Users\mjibr\Pictures\Saved Pictures` dengan nama file yang unik, berdasarkan counter yang ditambahkan setiap kali gambar disimpan.

- **Masalah yang Dihadapi:** Pada beberapa sistem, folder tujuan untuk menyimpan gambar harus memiliki izin akses. Selain itu, jika folder tidak ada, kita memastikan bahwa folder tersebut dibuat secara otomatis menggunakan `os.makedirs()`.

3. Kendala yang Dihadapi

Beberapa masalah atau kendala yang mungkin terjadi selama implementasi adalah:

- **Kesulitan dalam segmentasi warna:** Segmentasi warna adalah langkah penting dalam mendeteksi dokumen atau objek. Rentang warna yang salah atau pencahayaan yang buruk bisa menghambat proses deteksi.
- **Kesalahan dalam deteksi kontur dan perspektif:** Terkadang, kontur yang ditemukan tidak cukup akurat untuk melakukan transformasi perspektif. Hal ini bisa disebabkan oleh ketidakjelasan gambar atau objek yang terdistorsi.
- **Kinerja Webots:** Jika simulasi tidak berjalan dengan baik, pastikan robot diatur dengan benar dan tidak ada masalah dalam pengaturan dunia simulasi, seperti posisi kamera atau display.

4. Solusi dan Peningkatan

Beberapa solusi dan peningkatan yang dapat dilakukan untuk mengatasi masalah yang dihadapi:

- **Penyempurnaan Rentang Warna:** Gunakan alat seperti OpenCV atau GIMP untuk menyesuaikan rentang warna yang digunakan dalam segmentasi, agar lebih sesuai dengan warna dokumen yang akan dipindai.
- **Peningkatan Kualitas Gambar:** Jika gambar buram atau kurang jelas, pastikan robot ditempatkan di posisi yang benar, dengan pencahayaan yang cukup.
- **Pengujian Deteksi Perspektif:** Uji dengan berbagai jenis dokumen dan orientasi untuk memastikan bahwa `get_warped_document()` berfungsi dengan baik dalam semua kondisi.

5. Kesimpulan

Tugas ini berhasil menciptakan sistem pemindaian dokumen berbasis robot yang mampu mengoreksi perspektif gambar yang terdistorsi dan menampilkan tampilan **top-down** dari dokumen. Sistem ini menggunakan **image processing** dengan teknik segmentasi warna dan transformasi perspektif untuk memproses gambar dokumen yang diambil oleh kamera robot di **Webots**.

- **Kelebihan:** Sistem dapat menangani deteksi dan koreksi perspektif secara otomatis dengan memanfaatkan Webots dan OpenCV. Gambar yang telah diproses juga disimpan untuk analisis lebih lanjut.
- **Perbaikan yang Diperlukan:** Sistem perlu disesuaikan dengan pencahayaan yang baik dan kondisi dunia simulasi untuk memastikan deteksi dokumen berjalan dengan akurat.

Dengan penyesuaian lebih lanjut, sistem ini dapat digunakan untuk pemindaian dokumen dalam berbagai kondisi dan diterapkan di lingkungan dunia nyata.