

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Dalam penelitian ini, metode penelitian yang dilakukan adalah dengan studi literatur, perancangan, implementasi, dan pengujian sistem. Studi literatur adalah tahap metode penelitian yang dilakukan oleh peneliti dalam mengumpulkan referensi dan landasan teori mengenai masalah yang akan diteliti. Perancangan adalah tahap peneliti melakukan rancangan sistem untuk menyelesaikan masalah yang diangkat. Implementasi adalah tahap penerapan dari rancangan serta dilakukannya pengujian terhadap sistem, sehingga didapatkan kelebihan kekurangan dan hasil yang akan digunakan sebagai penilaian efektifitas dari sistem.

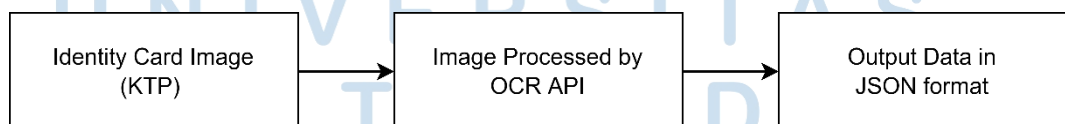
Studi literatur dilakukan oleh penulis dengan membaca, mempelajari referensi dan *research paper* mengenai OCR, segmentasi area, dan *text detector*. Bentuk rancangan sistem yang dilakukan oleh penulis dibahas pada BAB III. Implementasi dan pengujian sistem yang dirancang akan dibahas pada BAB IV.

3.2 Rancangan Sistem

3.2.1 Rancangan Umum

Berikut ini merupakan rancangan umum dari sistem pembacaan OCR pada Kartu Tanda Penduduk Indonesia dengan CRAFT dan Tesseract

OCR:

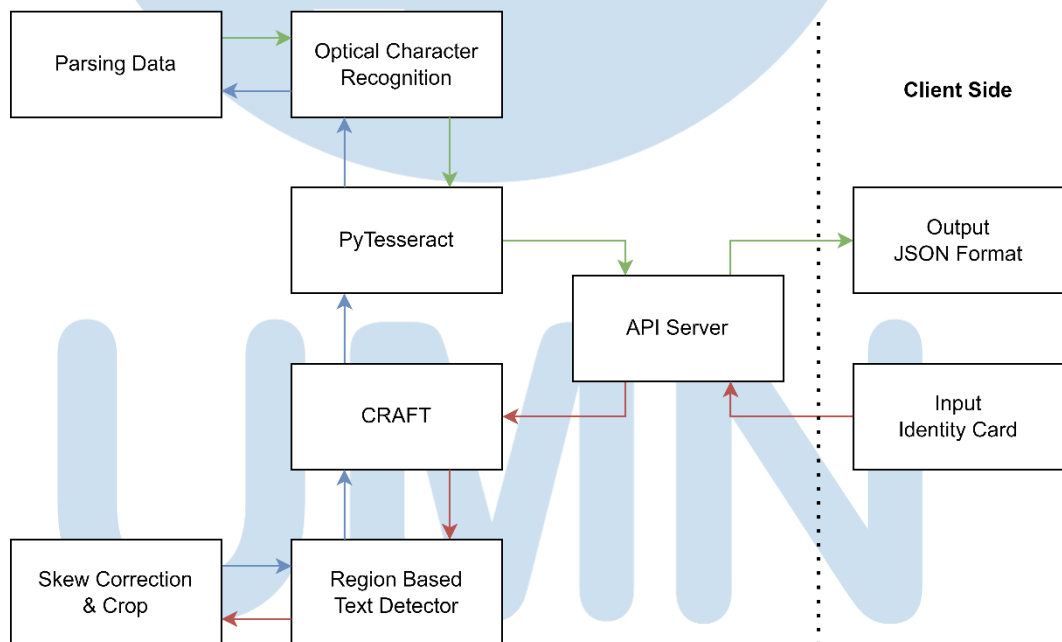


Gambar 3.1 Flowchart Rancangan Sistem

Pengguna mengakses API dengan mengirimkan input sistem berupa foto identitas yaitu Kartu Tanda Penduduk (KTP) yang datanya ingin diekstrak, lalu foto identitas akan diterima oleh API server kemudian diproses oleh OCR API sesuai Gambar 3.2, dan didapatkan output data berupa text dari beberapa field KTP dalam format JSON yang nantinya dapat digunakan untuk berbagai macam keperluan, sebagai contoh *form autofill*.

3.2.2 Komponen Sistem

Komponen detail sistem OCR akan dijalankan pada API yang dibuat dengan FastAPI. Alur proses OCR berjalan sebagai berikut:



Gambar 3.2 Flowchart komponen detail proses OCR

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Berikut ini merupakan penjelasan dari tiap komponen yang digunakan dalam sistem OCR API:

- Identity Card Photo

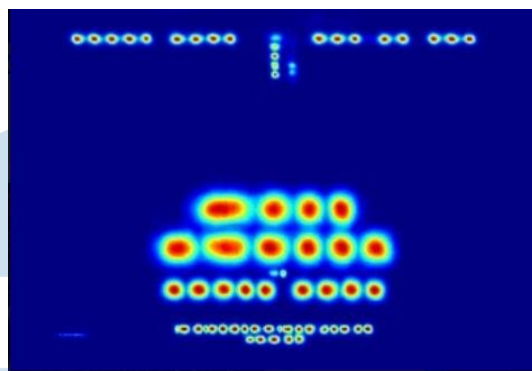
Input foto berupa Kartu Identitas berupa Kartu Tanda Penduduk (KTP) dikirimkan dari pengguna menuju server OCR API untuk diekstrak dan mendapatkan hasil berupa data text dari beberapa field KTP.

- Processing in CRAFT Text Detector

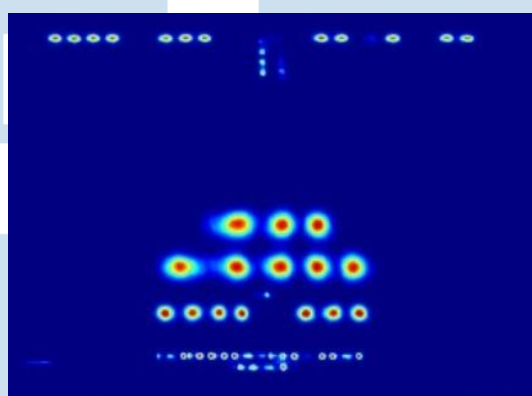
Gambar KTP kemudian diproses oleh CRAFT, proses ini terdiri dari 3 tahap, yaitu: pertama Region Map adalah tahap dimana text tiap karakternya pada gambar akan dideteksi dan ditandai seperti pada gambar 3.4, kedua Affinity Map adalah tahap dimana perhitungan relasi dari beberapa karakter yang sebelumnya didapatkan dari Region Map untuk membentuk suatu kata seperti pada gambar 3.5, artinya semakin tinggi Affinity dari susunan karakter tersebut maka susunan karakter tersebut memiliki relasi sebagai sebuah kata. Tahap ketiga, setelah lokasi tiap kata pada gambar telah diketahui CRAFT akan menggambar bounding box pada tiap kata tersebut, kemudian informasi koordinat dari 4 sudut bounding box akan disimpan.



Gambar 3.3 Contoh Input Gambar CRAFT [10]



Gambar 3.4 Region Map CRAFT



Gambar 3.5 Affinity Map CRAFT

- Skew Correction and Text Cropping

Ketika lokasi data tiap text pada KTP sudah diketahui selanjutnya proses yang dilakukan yaitu memperbaiki atau koreksi tingkat kemiringan pada gambar menggunakan fungsi `minAreaRect`, fungsi ini dipilih karena dasar dari skew correction yaitu *contour* atau garis tepi, kelemahannya yaitu jika terdeteksi *contour* diluar dari object yang ingin diperbaiki, `minAreaRect` akan *enclose* seluruh garis tepi teks pada KTP fungsi ini tepat sesuai dengan proses sistem dimana `craft` dapat dengan pasti mendeteksi teks KTP walau posisi tidak wajar dan didapatkan *contour* teks tersebut, dibandingkan dengan fungsi atau algoritma yang mendeteksi *contour* garis dari sisi luar persegi kartu yang dapat terpengaruhi oleh garis pada background. Proses skew correction ini akan menghasilkan gambar

menjadi lurus mendatar, dan juga dilakukan proses crop per kata pada gambar berdasarkan informasi koordinat yang didapatkan pada proses sebelumnya.

- **Image PreProcessing**

Proses image preprocessing penting dilakukan sebelum melakukan OCR, hal ini bertujuan agar OCR dapat mendeteksi gambar dengan baik tanpa gangguan dan akhirnya akan didapatkan hasil output yang lebih akurat. Tahap preprocessing ini terdiri atas mengurangi noise dan binarization yaitu mengubah warna pada gambar dalam bentuk binary hitam dan putih.

- **Processing in Tesseract OCR**

Gambar yang sebelumnya sudah dilakukan preprocessing sekarang sudah siap untuk dideteksi oleh Tesseract OCR untuk didapatkan output berupa data text. Untuk meningkatkan akurasi output dari bacaan KTP ditambahkan juga penggunaan trained data sesuai dengan jenis font yang digunakan pada KTP.



Gambar 3.6 Font Pada KTP [11]

- **Parsing Data**

Setelah didapatkan hasil output dari OCR, data tersebut berupa text dari beberapa field pada KTP, namun tidak semua teks dan

informasi dalam KTP dibutuhkan sehingga dilakukan proses eliminasi dan filtering dan akhirnya data siap dikirimkan dalam format JSON.

3.3 Rancangan Pengujian

Sistem pengujian hasil pembacaan OCR yang akan dilakukan yaitu berdasarkan evaluasi output menggunakan Character Error Rate (CER). Terdapat 3 jenis error yang dapat terjadi dalam hasil bacaan OCR, kesalahan ini akan digunakan untuk melakukan perhitungan CER.

- Substitution error : Kesalahan karakter/kata
- Deletion error : Kehilangan karakter/kata
- Insertion error : Penambahan karakter/kata



Gambar 3.7 Jenis-jenis error [12]

$$CER_{normalized} = \frac{S + D + I}{S + D + I + C}$$

Gambar 3.8 Rumus Perhitungan CER Normalized

S = Jumlah Substitutions

D = Jumlah Deletions

I = Jumlah Insertions

C = Jumlah karakter yang sesuai dengan reference text (ground truth)

Penggunaan CER sebagai benchmark pengujian dilakukan berdasarkan case yang berbeda-beda, sebagai contoh untuk kategori printed text dan

handwritten memiliki penilaian yang berbeda dimana untuk printed text didapatkan CER senilai 10% masih dianggap average, pada kategori handwritten penilaian average CER 20%. Berikut ini merupakan kategori penilaian CER untuk kategori printed text:

- Good OCR accuracy: CER 1-2% (i.e. 98–99% accurate)
- Average OCR accuracy: CER 2-10%
- Poor OCR accuracy: CER >10% (i.e. below 90% accurate)

