



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Metodologi Penelitian yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Studi Literatur

Pada tahap ini, akan dilakukan pencarian dan mempelajari setiap algoritma atau metode dan bahasa pemrograman yang dipakai dalam penelitian. *Optical Character Recognition, Artificial Neural Network, Convolutional Recurrent Neural Network, Convolutional Neural Network, Long Short Term Memory, Connectionist Temporal Classification* dan *Character Error Rate* merupakan metode atau algoritma yang digunakan dalam penelitian ini.

2. Pengumpulan Data

Data yang dipakai pada penelitian ini adalah berupa teks pada gambar yang merupakan *printed text* yang didapat dari *Text Recognition Data Generator*.

3. Perancangan dan Pembuatan Sistem

Merancang sistem dengan membuat *flowchart*, membuat program berbentuk *website* menggunakan bahasa *Python* dan *Flask*, dan mengimplementasikan semua algoritma atau metode yang digunakan ke dalam bentuk bahasa pemrograman yaitu, bahasa *Python* untuk melakukan tahap OCR yang didukung dengan hardware dan software lainnya.

4. Uji Coba dan Evaluasi

Pada tahap ini, akan dilakukan uji coba mengubah jumlah *batch size* dengan jumlah data yang berbeda. Setelah itu evaluasi yang dilihat dari *validation loss* saat *training* dan *Character Error Rate* dan akurasi saat *testing*.

5. Penulisan Skripsi

Tahap akhir dalam penelitian adalah pembuatan skripsi yang ditulis dan disusun dari awal hingga akhir sesuai dengan hasil penelitian. Laporan ini tersusun dari perancangan, pembuatan, hasil penelitian, serta dokumentasi selama penelitian berlangsung.

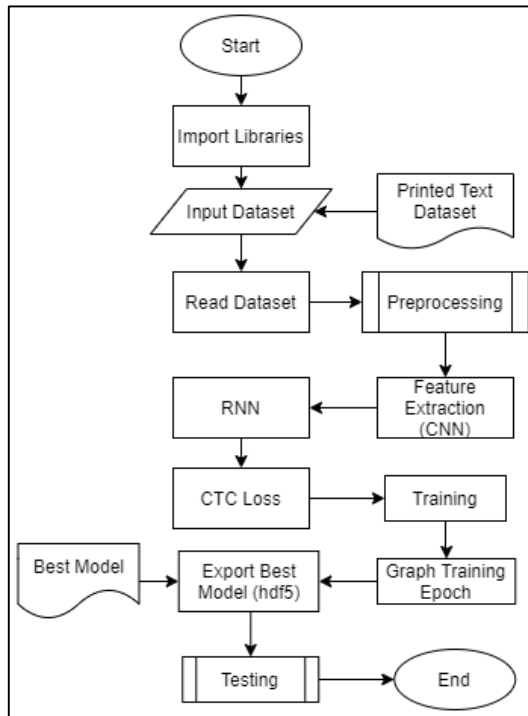
3.2 Perancangan Sistem

Perancangan sistem terdiri dari *flowchart* dan rancangan tampilan antarmuka pada aplikasi *website*. Berikut *flowchart* pada perancangan sistem.

3.2.1 Flowchart Proses OCR

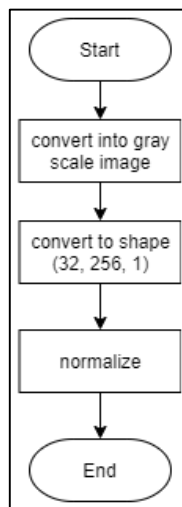
Pada Gambar 3.1. merupakan alur dari proses OCR pada penelitian menggunakan model CRNN secara umum. Tahap pertama merupakan *import libraries* yang dibutuhkan. Setelah itu, kemudian *dataset* masuk dan diproses. *Dataset* berupa gambar 1 kata *printed text* yang didapatkan dari *library trdg* atau *Text Recognition Data Generator* sejumlah total 160.000 dataset untuk *training*, *validation*, dan *testing*. *Dataset* langsung masuk ke tahap *preprocessing* yang dapat dilihat pada Gambar 3.2. Setelah itu *dataset* dapat masuk kedalam model CRNN yang dapat dilihat di Gambar 3.3. Lalu, *dataset* yang telah melewati proses dalam model memasuki tahap CTC Loss. Setelah itu dataset dapat di-*training*. Proses *Levenshtein Distance* untuk nanti diproses dan ditunjukkan pada saat memprediksi di *testing*. Setelah itu, hasil dari *training* di-*export* dan dipakai dalam *testing* dan

juga pada *website*. Tahap terakhir berupa testing yang dapat dilihat pada Gambar 3.4. Hasil keluaran dari prediksi atau *output* pada penelitian ini berupa *printed text*.



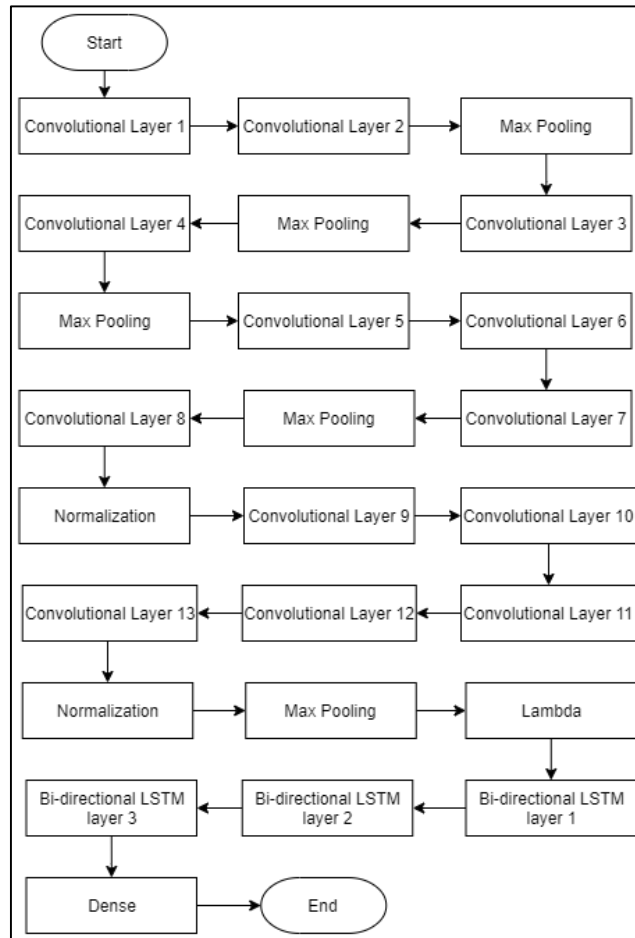
Gambar 3.1. *Flowchart* penelitian secara umum

Pada Gambar 3.2. ini merupakan *flowchart preprocessing* yang berisi 3 proses. *Dataset* akan di *convert* menjadi *gray scale image*, di-*convert* menjadi *shape* (32, 256, 1) agar dapat masuk kedalam model, dan juga *image* dinormalisasi.



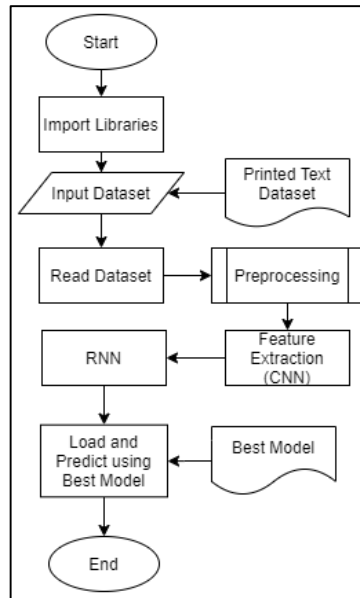
Gambar 3.2. *Flowchart Preprocessing*

Pada Gambar 3.3. merupakan proses Model CRNN yang arsitekturnya terinspirasi dari VGG-16 (Simonyan, 2015). *Dataset* akan masuk kedalam model dengan size (32, 256, 1) dan akan melewati 16 *layers* (13 *convolutional layers* dan 3 *bi-directional LSTM layers*), *max pooling*, *normalization*, *lambda*, dan *dense*.



Gambar 3.3. *Flowchart Model CRNN*

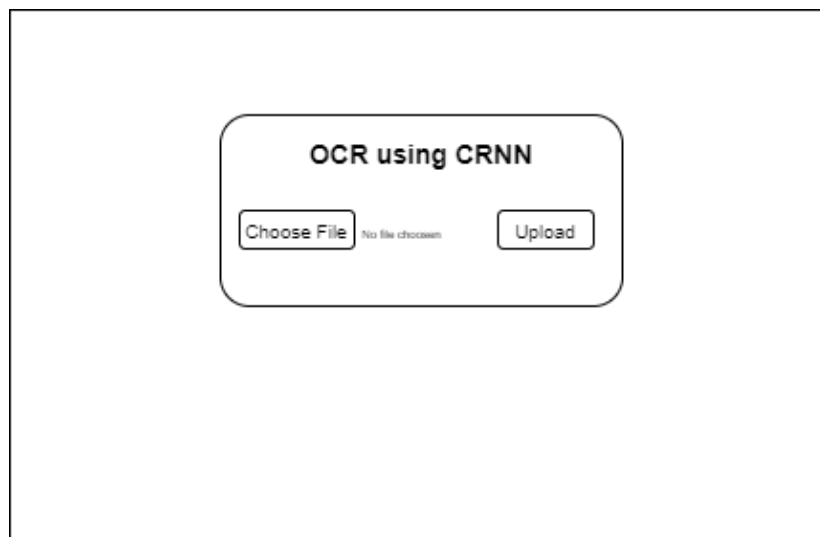
Pada gambar 3.4. merupakan proses *testing* dari hasil yang sudah di selesai *di-training*. Dalam proses *testing*, *levenshtein distance* berperan untuk melihat total kata yang berubah dengan membandingkan *original text* dengan kata hasil prediksi.



Gambar 3.4. *Flowchart Testing*

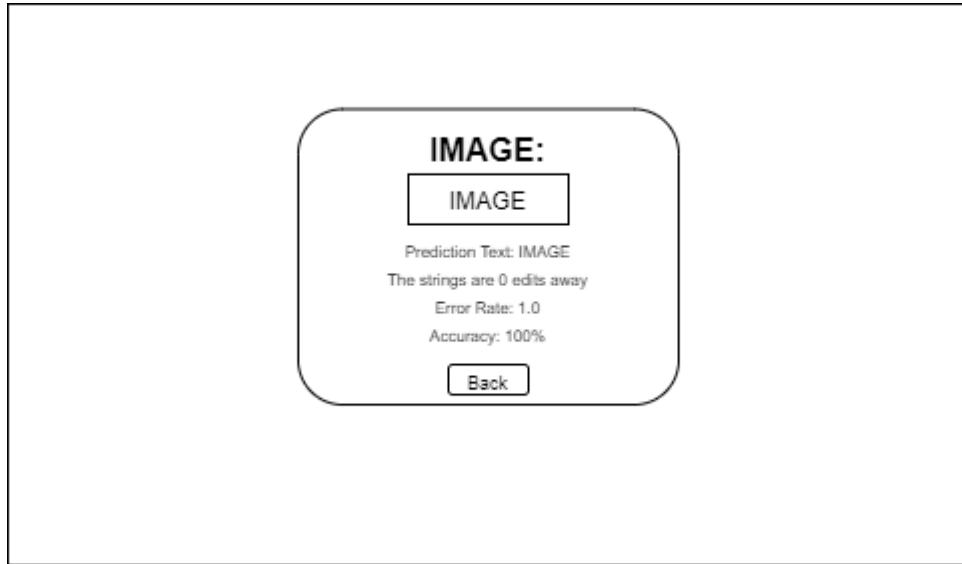
3.2.2 Rancangan Tampilan Antarmuka

Tampilan antarmuka aplikasi *website* terbagi menjadi dua halaman, yaitu halaman utama dan halaman hasil. Pada Gambar 3.5. merupakan rancangan antarmuka untuk halaman utama. Pada halaman ini *user* dapat *upload file* gambar yang akan diprediksi.



Gambar 3.5. Rancangan Halaman Utama Aplikasi

Pada Gambar 3.6. merupakan rancangan halaman hasil aplikasi *website*. Pada halaman ini, *user* dapat melihat hasil prediksi dari file gambar yang telah di upload dihalaman sebelumnya.



Gambar 3.6. Rancangan Halaman Hasil Aplikasi