**BAB III**

**METODOLOGI PENELITIAN**

**3.1 Pengumpulan Data**

Data yang digunakan dalam penelitian ini adalah citra kemasan biskuit Nextar yang baik dan rusak. Data baik diambil dari sampel produksi yang dianggap baik dan data rusak diambil dari sampel produksi yang mengalami kerusakan kemasan. Data yang diperoleh akan dibersihkan dan diproses pada tahap berikutnya. Berikut Contoh sampel data yang diambil untuk diolah :



Gambar 3.1 Tampilan Kemasan Yang bagus



Gambar 3.2 Tampilan Kemasan yang rusak

**3.2 Preprocessing Data**

Pada tahap ini, data yang diperoleh akan dibersihkan dan diproses untuk menghilangkan noise dan memperbaiki kualitas citra. Preprocessing data meliputi konversi citra ke dalam format grayscale, normalisasi citra, serta resizing citra untuk menyesuaikan ukuran citra yang diinginkan.

Preprocessing pada salah satu sampel data terkait label produk yang bagus dan rusak :



Gambar 3.3 Preprocessing Data yang diambil untuk ditraining

**3.3 Pembuatan Model**

Model Deep Learning akan dibangun menggunakan *Tensorflow* dan CNN. Pembuatan model akan dilakukan dengan mengatur parameter seperti jumlah lapisan (layers), fungsi aktivasi, dan pengaturan lainnya. Model akan dilatih menggunakan data baik dan rusak yang telah dipreproses. Pada penelitian ini, model CNN yang digunakan terdiri dari beberapa lapisan, yaitu:

- Lapisan Input: berfungsi sebagai input data gambar

- Lapisan Konvolusi: berfungsi untuk mengekstrak fitur-fitur dari gambar menggunakan kernel konvolusi

- Lapisan ReLU: berfungsi untuk mempercepat proses pelatihan dengan menghilangkan nilai negatif pada lapisan konvolusi

- Lapisan Max Pooling: berfungsi untuk mengurangi dimensi dari hasil konvolusi dan mempertahankan fitur yang paling penting

- Lapisan Dropout: berfungsi untuk mencegah overfitting dengan menonaktifkan beberapa neuron secara acak selama pelatihan

- Lapisan Flatten: berfungsi untuk meratakan hasil pooling menjadi vektor fitur.

- Lapisan Dense: berfungsi untuk menghubungkan vektor fitur dengan kelas output

- Lapisan Output: berfungsi sebagai output hasil klasifikasi

**3.4 Pengujian Model**

Setelah model selesai dilatih / *train* akan dilakukan proses pengujian model. Dengan menggunakan Python sebagai platform untuk mengevaluasi hasil model yang telah dilatih. Dengan menggunakan library seperti NumPy dan Pandas untuk membaca dan memuat dataset. Selanjutnya, dapat menggunakan TensorFlow dan Keras untuk membangun model CNN dan melatih model dengan dataset yang dimuat agar dapat diteruskan untuk mengevaluasi performa model menggunakan metrik evaluasi seperti berikut :

**a.Akurasi**

Akurasi adalah ukuran persentase dari jumlah klasifikasi yang benar dibandingkan dengan total klasifikasi yang dilakukan. Akurasi dihitung dengan membagi jumlah klasifikasi yang benar dengan jumlah total klasifikasi.

**b.Presisi**

Presisi adalah ukuran persentase dari jumlah hasil klasifikasi yang benar positif dibandingkan dengan total hasil klasifikasi positif. Presisi dihitung dengan membagi jumlah hasil klasifikasi benar positif dengan jumlah total hasil klasifikasi positif.

**c. Recall**

Recall adalah ukuran persentase dari jumlah hasil klasifikasi yang benar positif dibandingkan dengan total jumlah data yang seharusnya diklasifikasikan positif. Recall dihitung dengan membagi jumlah hasil klasifikasi benar positif dengan jumlah total data yang seharusnya diklasifikasikan positif.

**d. F1-score**

F1-score adalah rata-rata harmonik dari presisi dan recall. F1-score memberikan ukuran yang lebih seimbang antara presisi dan recall daripada hanya menggunakan satu dari dua metrik ini saja.

Dalam pengujian ini output dari python hanya sebatas data output label dan akurasi yang ditampilkan masih dalam bentuk data *float* dan memerlukan sebuah library lagi bernama *Flask* untuk membuat data uji menjadi *Backend* *Restful API* supaya bisa diolah lagi oleh *frontend* website agar mudah digunakan oleh pengguna. Agar aplikasi berjalan lancar dengan keamanan yang baik maka diperlukan keamanan yang lebih dengan menggunakan text kunci agar tidak semua orang bisa mengakses datanya.

**3.5 Penerapan Model**

Setelah selesai membuat *backend Rest API* untuk menguji data model maka dibutuhkan *frontend* agar pengujian model bisa dilakukan oleh user dengan nyaman. Dalam penelitian ini mengambil framework *javascript vue* sebagai *frontend*. *Vue* adalah salah satu framework JavaScript yang dapat digunakan untuk membangun aplikasi web interaktif dengan cepat dan mudah. Dalam konteks penggunaannya sebagai *frontend* dari *Deep Learning* menggunakan *Tensorflow* dan *Convolutional Neural Network* untuk pendeteksian kemasan biskuit Nextar yang rusak, Vue dapat mempermudah penggunaan dan integrasi antara model Deep Learning dan tampilan aplikasi web.

Dalam proses pendeteksian kemasan biskuit Nextar yang rusak, model Deep Learning yang dibuat menggunakan Tensorflow dan Convolutional Neural Network digunakan untuk memproses gambar kemasan biskuit dan menghasilkan prediksi apakah kemasan biskuit tersebut rusak atau tidak. Kemudian, hasil prediksi tersebut ditampilkan pada tampilan aplikasi web yang dibangun dengan Vue.

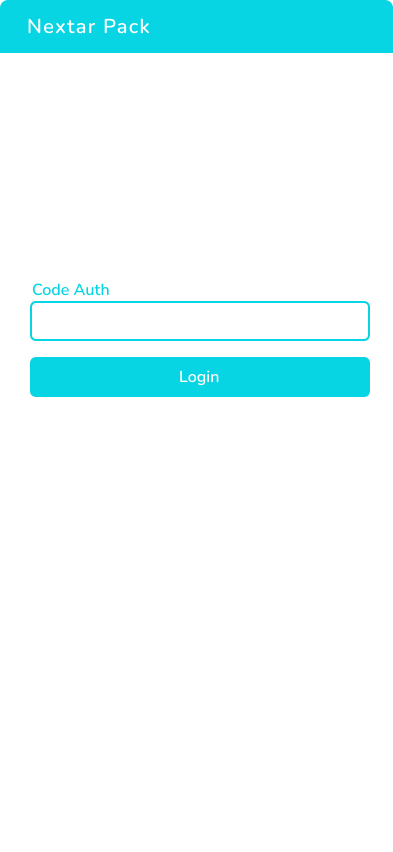
Dengan menggunakan Vue sebagai frontend, pengguna dapat dengan mudah mengakses dan memanipulasi hasil prediksi dari model Deep Learning, serta menampilkan informasi tambahan seperti gambar kemasan biskuit yang diproses dan hasil prediksi yang dihasilkan. Hal ini dapat meningkatkan efektivitas dan efisiensi dalam proses pendeteksian kemasan biskuit yang rusak.

Selain itu, Vue juga memiliki kemampuan untuk mengintegrasikan komponen-komponen lain seperti grafik dan tabel, yang dapat membantu pengguna dalam memvisualisasikan dan menganalisis hasil prediksi dari model Deep Learning dengan lebih mudah dan cepat.

Dalam pengembangan penelitian ini dengan menggunakan system yang tergolong simpel ini diperlukan beberapa halaman ini agar keamanan dan kenyamanan pengguna terlaksana. Struktur halamannya sebagai berikut :

**3.5.1 Login Form**

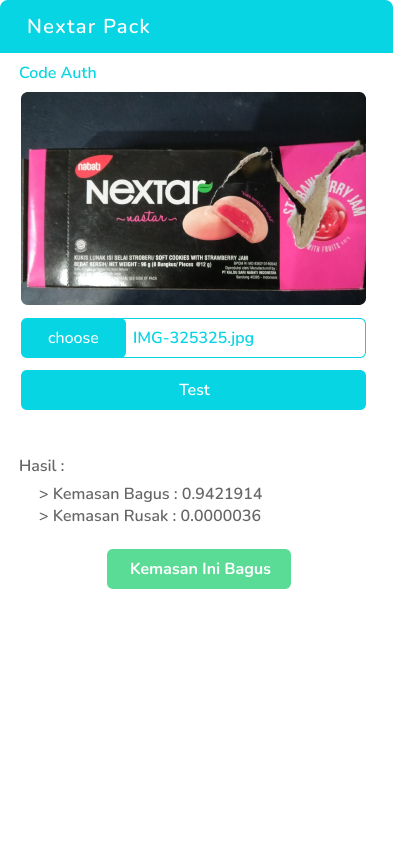
Dalam Login Form ini menggunakan key untuk dikirim kepada backend pythonnya agar mendapakan autentikasi untuk mendapatkan akses data. Login Form ini berfungsi agar tidak semua orang dapat mengakses data.



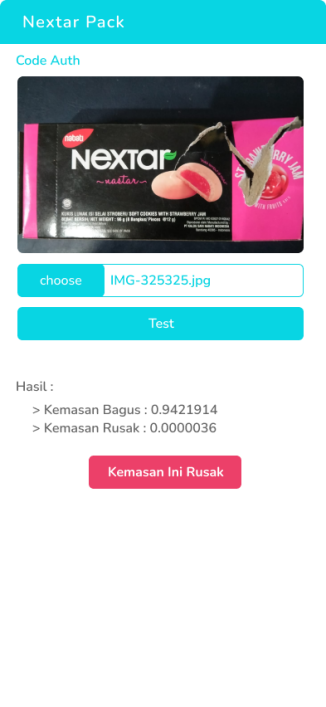
Gambar 3.3 Login Form

**3.5.2 Halaman Tes Uji**

Dalam halaman tes uji ini berisi mengenai hal hal yang diperlukan dalam melakukan tes uji. Ada gambar yang harus dikirim untuk dilakukan tes uji apakah gambar tersebut merupakan kemasan nextar yang bagus apa yang rusak.



Gambar 3.4 Hasil tes uji apabila kemasan bagus



Gambar 3.5 Hasil tes uji apabila kemasan rusak