

PERANCANGAN MOBILE APPLICATION UNTUK MENGKLASIFIKASIKAN SAYUR SEGAR DAN BUSUK MENGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)

Rizky Bintang Sahputra, Anang Kunaefi, Andhy Permadi

Sistem Informasi, Universitas Islam Negeri Sunan Ampel Surabaya

Jl. Dr. Ir. H. Soekarno No.682, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur 60294

rbintang447@gmail.com

ABSTRAK

Sayuran merupakan sebuah sumber nutrisi dan gizi utama bagi manusia. Sayur memiliki beragam jenisnya, seperti bayam, kangkung, wortel, timun, dan tomat. Sayur memiliki beragam manfaat untuk menjaga kesehatan tubuh manusia. Namun disamping melihat beragam manfaat yang dimilikinya, tentunya hal itu tidak akan berguna apabila sayur tersebut tidak segar atau bisa dikatakan busuk. Sehingga jika dikonsumsi oleh manusia bisa menjadi sebuah penyakit. Maka dari itu, dengan menggunakan perkembangan teknologi dan AI, muncul inovasi yang bisa digunakan, yaitu aplikasi deteksi sayur dengan menggunakan metode Convolutional Neural Network (CNN). Penelitian ini dibuat dengan tujuan untuk membantu masyarakat untuk memilih sayur mana yang punya kualitas segar dan busuk. Sehingga bisa mengurangi resiko terkena penyakit ketika mengkonsumsi makanan busuk. Platform yang dipakai untuk merancang aplikasi pada penelitian ini adalah android. Jenis sayur yang dipakai pada penelitian ini terdapat 5 jenis, yaitu timun, wortel, brokoli, selada, dan kubis. Dimana terdapat total 400 dataset yang didapat dengan menggunakan metode observasi dan wawancara. Dimana 90% dataset digunakan untuk train model dan sisanya untuk pengujian. Pengujian dilakukan dengan menggunakan 10 sampel Dimana satu sampel dilakukan uji sebanyak 5 kali. Hasil pengujian model berhasil mendapat angka 86% akurasi dalam mengklasifikasikan sayur segar dan busuk.

Kata kunci : *Deep Learning, Convolutional Neural Network, Klasifikasi*

1. PENDAHULUAN

Sayuran merupakan sebuah sumber nutrisi dan gizi utama bagi manusia. Sayur sendiri memiliki banyak kandungan zat gizi, serat, vitamin, mineral dan zat yang berfungsi untuk menjaga kesehatan tubuh manusia agar tetap sehat dan bugar [1]. Sayur memiliki beragam jenisnya, seperti bayam, kangkung, wortel, timun, dan tomat. Semua jenis sayur itu memiliki beragam manfaat yang bagus untuk tubuh manusia seperti meningkat imunitas tubuh, menjaga kesehatan mata, membantu memperlancar sistem pencernaan tubuh. Sehingga disini mengkonsumsi sayuran itu penting bagi manusia mengingat beragam manfaat yang dimilikinya. Namun disamping melihat beragam manfaat yang dimilikinya, tentunya hal itu tidak akan berguna apabila sayur tersebut tidak segar atau bisa dikatakan busuk. Sehingga bukan hanya kandungan nutrisinya yang berkurang, jika dikonsumsi oleh manusia bisa menjadi sebuah penyakit.

Ciri-ciri yang paling umum dari sayur segar sendiri adalah warna cerah tajam, tidak layu, ukuran kecil dan muda, serta aroma yang alami [2]. Di sisi lain, sayur yang tidak segar atau mulai busuk, memiliki warna pucat, layu, lembek, dan aroma yang tidak sedap. Namun, indikator ciri-ciri kadang-kadang tidak benar sepenuhnya; misalnya, sayuran mungkin memiliki ciri-ciri membusuk tetapi tidak membusuk. Jadi, ilmu, pengalaman, dan ketelitian diperlukan untuk mengetahui apakah sebuah sayur segar atau tidak.

Untuk membantu hal ini, aplikasi deteksi sayur dengan teknologi Convolutional Neural Network

(CNN) digunakan untuk mendeteksi dan mengidentifikasi kesegaran sayuran secara otomatis. CNN adalah teknik kecerdasan buatan yang cocok untuk digunakan dalam menganalisis gambar visual. Dimana metode CNN ini nantinya akan menemukan pola penting yang menunjukkan tingkat kesegaran sayuran. CNN dapat membedakan warna cerah pada sayur yang masih segar dan warna kekuningan yang menunjukkan penurunan kualitas kesegaran sayur, atau membedakan tekstur halus pada daun yang masih segar dan yang sudah layu.

Teknologi CNN memproses gambar sayuran yang diambil dari kamera atau perangkat smartphone dalam beberapa tahap. Selama proses ini, sistem dapat mengekstrak karakteristik seperti tekstur, warna, dan bentuk sayuran. Data ini kemudian digunakan untuk mengklasifikasikan apakah sayuran tersebut masih segar atau sudah mengalami penurunan kualitas. Aplikasi juga memfasilitasi pengumpulan data besar melalui Big Data Analytics, Hal ini membuat informasi seluruh sayur pada pasar tradisional atau modern yang ada bisa dimuat. Aplikasi ini memberikan analisis mengenai data sayuran segar di suatu pasar tiap minggunya. Sehingga masyarakat bisa melihat pasar yang memiliki persentase sayur segar yang tinggi.

Dengan demikian, penelitian ini nantinya diharapkan bisa membantu konsumen dalam memilih sayuran berkualitas, serta meningkatkan minat Masyarakat dalam mengkonsumsi sayur. Hal ini dilakukan supaya tercipta pola hidup sehat. Sehingga diharapkan penelitian ini dapat berkontribusi pada

peningkatan kesehatan masyarakat secara keseluruhan.

2. TINJAUAN PUSTAKA

Jika melihat dari penelitian sebelumnya [3] metode CNN juga pernah digunakan mengklasifikasi penyakit pada cabai. Pada penelitian ini diperoleh 1000 gambar yang diambil pada Desa Savabajaya. Dari 1000 gambar tersebut dibagi lagi menjadi dua, 500 dataset cabai sehat dan 500 dataset cabai terinfeksi. Konfigurasi optimizer yang digunakan pun menggunakan adam untuk menghasilkan akurasi tinggi sekitar 93.25%. Penelitian ini dibuat dengan tujuan untuk membantu petani sekitar untuk mengendalikan penyakit pada cabai. Dimana hal ini akan berdampak pada produktivitas petani dan mengurangi kerugian yang diakibatkan penyakit.

Kemudian pada penelitian [4] metode CNN Kembali digunakan untuk mendeteksi penyakit pada tomat. Penelitian ini menggunakan platform android untuk pengimplementasian aplikasinya. Pada penelitian ini akurasi yang dihasilkan memiliki angka yang besar, yaitu 94%. Namun, Ketika diimplementasikan ke dalam mobile aplikasi akurasi berkurang menjadi 92.80%. Akurasi ini diperoleh dengan menggunakan konfigurasi Adam.

2.1. Deep Learning

Deep Learning merupakan salah satu dari banyaknya cabang ilmu dari machine learning. Algoritma dari deep learning ini terinspirasi dari jaringan otak manusia atau bisa disebut dengan ANN (Artificial Neural Network). Deep learning ini mempunyai beberapa algoritma penting didalamnya, seperti Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), Recurrent Neural Network (RNN), Self Organizing Maps (SOM) [5].

2.2. CNN (Convolutional Neural Network)

CNN atau lebih dikenal dengan Convolutional Neural Network merupakan metode yang digunakan khusus untuk memproses data seperti pola dan gambar. CNN sendiri merupakan teknik yang ada dalam deep learning. CNN mempunyai hasil akurasi yang tinggi, karena CNN mengekstrak fitur penting yang ada dalam suatu gambar, sehingga detail-detail dari suatu visual bisa tertangkap dengan baik. Dengan adanya kombinasi fitur-fitur tersebut bisa membuat CNN semakin akurat dalam mengenali dan mengklasifikasi sesuatu [6].

2.3. Klasifikasi Gambar

Klasifikasi gambar bertugas untuk mengelompokkan gambar yang sudah diambil ke dalam beberapa kelas. Dalam penelitian ini, klasifikasi gambar dibagi menjadi dua kelas, yaitu sayur segar dan sayur busuk. Klasifikasi ini dilakukan dengan menggunakan metode CNN dalam pemrosesannya.

2.4. Mobile Application

Mobile Application atau lebih dikenal dengan mobile apps merupakan sebuah perangkat lunak yang dirancang dan dibuat khusus untuk smartphone. Mobile apps ini didesain supaya bisa dijalankan pada platform android dan ios. Mobile apps ini punya beragam jenis aplikasi mulai dari permainan, media sosial, toko online [7]. Mobile apps ini juga dirancang supaya aplikasi yang dibuat bisa diakses dengan mudah dimanapun dan kapanpun dengan leluasa. Sehingga pengguna bisa menggunakannya secara fleksibel.

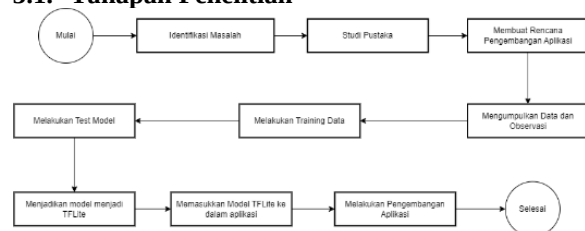
2.5. Kotlin

Kotlin merupakan Bahasa pemrograman yang digunakan untuk proses pengembangan aplikasi mobile android. Bahasa pemrograman ini didukung oleh google dan bisa digunakan pada software android studio. Bahasa pemrograman Kotlin bersifat open source dan memiliki berbagai fitur yang mudah digunakan. Bahasa pemrograman kotlin berjalan pada platform JVM dan memakai compiler LLVM yang dapat dikompilasi ke javascript [8].

3. METODE PENELITIAN

Studi ini menggunakan metode Convolutional Neural Network untuk mengklasifikasikan sayuran menjadi 2 bentuk, yaitu sayur segar dan sayur tidak segar. Proses pengklasifikasian ini menggunakan Bahasa pemrograman python menggunakan metode CNN. Peneliti juga menggunakan sistem berbasis Android untuk perancangan aplikasi. Dalam penelitian ini, metode pengembangan sistem yang digunakan adalah waterfall. Namun sebelum menuju pada semua metode tersebut, pertama-tama terdapat alur berjalannya penelitian ini yang bisa dilihat pada Gambar 1.

3.1. Tahapan Penelitian



Gambar 1. Alur Penelitian

Berikut merupakan penjelasan dari Gambar 1 yang menjelaskan tentang tahapan alur dari penelitian ini.

a. Identifikasi Masalah

Pada tahap pertama dimulai dengan mengidentifikasi masalah dari penelitian yang akan dilakukan. Dimana kendala pada penelitian ini semuanya ditulis dan dijelaskan satu persatu. Mulai dari kebutuhan pengguna hingga masalah yang terjadi selama proses penelitian. Sehingga semua masalah yang ada harus bisa diselesaikan setelah penelitian ini selesai.

b. Studi Pustaka

Tahap kedua dari alur penelitian ini yaitu, melakukan studi pustaka dengan melihat penelitian terdahulu yang hamper mirip. Studi Pustaka ini dilakukan dengan melihat jurnal dan artikel yang memiliki kesesuaian dengan judul penelitian. Hal ini dilakukan untuk menambah wawasan peneliti tentang penelitian yang akan dilakukan. Sehingga bisa menghindari kemungkinan yang terburuk selama proses penelitian.

c. Membuat Rencana Pengembangan Aplikasi

Tahap ketiga merupakan membuat rencana untuk mengembangkan aplikasi. Pada tahap ini beragam deskripsi tentang hal yang berhubungan pengembangan aplikasi semua dijabarkan dengan jelas. Mulai dari fiturnya, alur pengembangan aplikasi, kebutuhan pengguna, serta semua teknologi yang akan digunakan dalam aplikasi yang akan dibuat.

d. Mengumpulkan Data dan Observasi

Tahap keempat lanjut untuk mengumpulkan data dengan observasi langsung ke beberapa pasar. Karena keterbatasan waktu, peneliti memutuskan untuk mengumpulkan data hanya di pasar yang berada di Surabaya. Proses pengumpulan data dilakukan dengan menggunakan kamera smartphone.

e. Melakukan Training Data

Data yang sudah dikumpulkan sebelumnya dibagi menjadi dua, satu untuk train dan satu lagi untuk test. Kemudian tahap selanjutnya yaitu, melakukan train data dengan menggunakan program python yang sudah dibuat. Hal ini dilakukan untuk membuat model yang akan digunakan pada aplikasi nanti bisa mempunyai akurasi dan keakuratan tinggi. Terkadang hasil train bisa berubah-ubah, maka dari itu, peneliti memilih hasil model yang paling bagus diantara semua train yang sudah dilakukan.

f. Melakukan Test Model

Setelah proses train model, tahap selanjutnya yaitu, mengetes model yang sudah dipilih sebelumnya. Dimana model yang sudah dipilih ini merupakan hasil terbaik selama proses train sebelumnya. Tahap ini ada untuk menguji model yang sudah dibuat sebelumnya, apakah model sudah bekerja dengan baik dan akurasi nya sudah sesuai. Sehingga performa dari aplikasi bisa digunakan tanpa hambatan.

g. Menjadikan Model Menjadi TFLite

Ketika model yang dipilih sebelumnya sudah cocok dan sesuai dengan standar yang diinginkan, maka bisa lanjut ke tahap ini. Dimana model tersebut akan dikonversi menjadi model TFLite. Hal ini perlu dilakukan supaya modelnya bisa tersimpan dan dimasukkan ke dalam basis android.

h. Memasukkan Model TFLite ke Dalam Aplikasi

Setelah mengkonversi model menjadi TFLite, tahap selanjutnya yaitu, memasukkannya ke dalam aplikasi yang akan dibuat. Model TFLite ini

nantinya akan digunakan untuk memprediksi gambar yang difoto menggunakan aplikasi.

i. Melakukan Pengembangan Aplikasi

Pada tahap terakhir dimana semua bahan sebelumnya sudah siap, tinggal mengembangkan aplikasi yang akan digunakan nantinya. Nantinya fitur aplikasi dan desain UI/UX akan diimplementasikan sesuai dengan requirement yang didapat sebelumnya. Tidak hanya itu, setelah selesai diimplementasikan, aplikasi diuji secara keseluruhan untuk memastikan fungsi dari aplikasi bisa berjalan dengan baik sepenuhnya.

3.2. Metode Pengumpulan Data

a. Studi Pustaka

Tahap pertama yang dilakukan untuk mengumpulkan data pada penelitian ini, yaitu dengan melakukan studi pustaka. Dimana peneliti melihat dan mencari informasi mengenai topik penelitian yang mirip sebelumnya pada sebuah jurnal dan artikel. Hal ini dilakukan supaya peneliti bisa mencari mana yang kurang dari penelitian sebelumnya dan lebih meningkatkan lagi pada penelitian ini. Selain mencari perbedaan dari penelitian terdahulu, peneliti tentunya juga bisa tahu apa saja langkah yang dilakukan untuk proses penelitian yang hampir serupa.

b. Observasi

Selain pengumpulan data secara studi pustaka, peneliti juga melakukan observasi secara langsung pada pasar-pasar yang menjual sayur. Dengan observasi juga, peneliti bisa mendapat data gambar sayuran yang akan dibutuhkan nantinya. Sehingga selain melihat kondisi berbagai macam sayuran di beberapa pasar di Surabaya, peneliti juga bisa mendapatkan data.

c. Wawancara

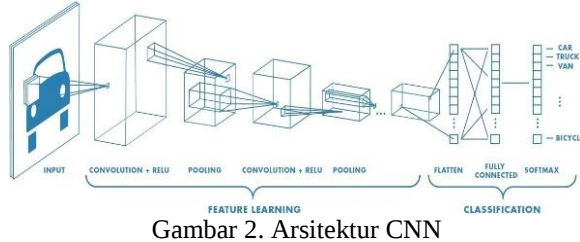
Selain kedua cara diatas, peneliti juga menggunakan metode wawancara untuk mencari informasi mengenai sayur mana yang sekiranya segar dan mulai membusuk. Sehingga pada saat proses pemodelan ini tidak terjadi kesalahan dalam pelabelan data. Apabila hal ini terjadi kesalahan nantinya hasil model yang dihasilkan akan fatal akibatnya. Selain mencari informasi mengenai sayur segar dan busuk, peneliti juga mencari requirement kepada pengguna. Sehingga nantinya aplikasi yang akan dibuat bisa digunakan dengan baik dan sesuai dengan pengguna.

3.3. Kebutuhan Data Penelitian

Pada penelitian ini menggunakan beragam data sekitar 400 gambar sayuran di beberapa pasar. Semua gambar tersebut juga dibagi lagi menjadi berbagai jenis, yaitu timun, wortel, kubis, brokoli, selada. Dari 400 gambar tersebut dibagi lagi menjadi 3 bagian, sesuai dengan pasar yang diteliti. Jadi tiap satu pasar terdapat sekitar 130 data gambar sayur. Serta dari 400 data gambar tersebut dibagi lagi jadi dua, 90% data

untuk proses train model CNN dan 10% sisanya untuk proses testing model.

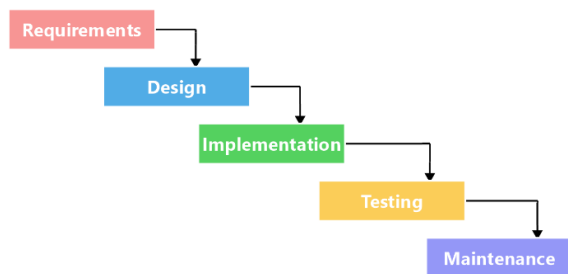
3.4. CNN



Gambar 2. Arsitektur CNN

Dari Gambar 2 algoritma CNN terbagi menjadi 2 lapisan, yaitu Classification dan Feature Learning. Sama seperti namanya, bagian Classification ini berfungsi untuk mengklasifikasikan data yang sudah disiapkan sebelumnya berdasarkan pola atau tepi [9]. Bisa dilihat juga pada lapisan classification ini dibagi menjadi 3 lapisan lagi didalamnya, yaitu Flatten, Fully Connected, dan Softmax. Sedangkan Feature Learning berfungsi untuk mengubah input menjadi fitur-fitur berdasarkan karakteristik unik masing-masing [9]. Proses ini dilakukan oleh lapisan-lapisan didalamnya, yaitu Convolutional Layer dan Pooling Layer. Hasil dari kedua lapisan ini nantinya akan dipakai dalam proses pembuatan aplikasi berbasis android [10].

3.5. Waterfall



Gambar 3. Urutan Model Waterfall

Berikut penjelasan urutan model waterfall diatas pada penelitian yang akan dilakukan, sesuai dengan Gambar 3.

- Requirement Analysis (Analisis Kebutuhan).** Tahapan yang ada pada requirement analysis digunakan untuk mengumpulkan data tentang kebutuhan yang diinginkan oleh pengguna pada aplikasi yang akan dibuat nanti. Hasil yang diperoleh nanti akan berupa analisis spesifikasi pada aplikasi yang akan dibuat [11]. Jadi pada tahap ini akan menghasilkan sebuah requirement dari pengguna yang harus diterapkan pada sistem informasi yang akan dibuat.
- Tahap kedua dari model waterfall adalah design.** Pada tahap ini dibuat sebuah desain yang digunakan untuk memberi sebuah gambaran mengenai sistem yang akan dibangun. Desain ini meliputi pembuatan desain arsitek, diagram UML (Use Case dan Activity Diagram), alur berjalannya

sistem, serta desain CDM, PDM, dan ERD dari basis data aplikasi [12].

- Tahapan ketiga dari model waterfall yaitu, implementation.** Pada tahap ini sistem sudah mulai dirancang dan dibangun sesuai dengan desain yang sudah dibuat sebelumnya pada tahap 2 model waterfall. Proses perancangan sistem akan melalui beberapa tahapan-tahapan dalam bentuk modul kecil yang ada pada metode waterfall. Dimana selanjutnya modul-modul kecil yang sudah ada tadi akan digabungkan jadi satu kesatuan. Seperti yang sudah dijelaskan di atas setiap modul harus dicek apakah sudah memenuhi fungsi dan sesuai dengan standar sebelumnya [13].
- Tahap selanjutnya adalah testing,** merupakan tahap keempat dari model waterfall. Pada tahap ini sistem yang sudah selesai dibuat akan dilakukan tes pengujian. Tahap ini penting untuk dilakukan karena tiap fitur dan fungsi yang dikembangkan pada tahap implementasi akan diintegrasikan ke dalam sistem. Setelah diintegrasikan seluruh sistem tersebut akan diuji untuk mengecek setiap kegagalan maupun kesalahan yang terjadi sebelum diimplementasikan ke suatu instansi [14].
- Tahap yang terakhir dari model waterfall ini yaitu, maintenance.** Pada tahap ini dilakukan sebuah pengoptimalan aplikasi, agar dapat berjalan dengan baik ketika diimplementasikan ke suatu instansi. Tidak hanya itu, apabila hasil dari testing sebelumnya terjadi bug dalam pengujian aplikasi, maka pada tahap ini akan dilakukan perbaikan bug dari sistem tersebut [12]. Kelima alur ini akan kembali lagi ke awal apabila user ingin menambahkan sebuah fitur baru pada sistem tersebut. Sehingga akan terjadi siklus perulangan pada proses pengembangan aplikasi untuk meningkatkan performa dari aplikasi atau sistem yang dibuat.

4. HASIL DAN PEMBAHASAN

Dibawah ini terdapat hasil dan pembahasan untuk perancangan mobile aplikasi deteksi sayur menggunakan metode Convolutional Neural Network (CNN) untuk sistem deteksi sayuran. Supaya sistem yang akan dikembangkan ini bisa berfungsi dengan baik dalam penggunaannya, maka perlu sebuah spesifikasi perangkat berupa software dan hardware yang mendukung perancangan mobile aplikasi tersebut. Dibawah ini beberapa perangkat lunak dan perangkat keras yang digunakan selama proses perancangan mobile aplikasi deteksi sayur.

4.1. Perangkat Lunak

Untuk membuat sebuah sistem memerlukan sebuah perangkat lunak. Berbagai perangkat lunak yang dipakai pada penelitian ini bisa dilihat pada Tabel 1.

Tabel 1. Perangkat Lunak

No	Software	Tipe Software
1	Sistem Operasi	Windows 11 Office Home and Student 64-bit
2	Bahasa Pemrograman	Python, Kotlin
3	Framework	Flutter
4	Module	Tensorflow CV2 Numpy Matplotlib

4.2. Perangkat Keras

Sama seperti sebelumnya, perangkat lunak ada dan digunakan untuk mendukung proses pembuatan sistem. Begitu juga dengan perangkat keras, tanpa sebuah perangkat keras sistem tidak akan bisa dibuat dan berbagai macam perangkat lunak tadi tidak akan bisa dipakai. Dibawah ini perangkat keras yang digunakan pada penelitian ini bisa dilihat pada Tabel 2.

Tabel 2. Perangkat Keras

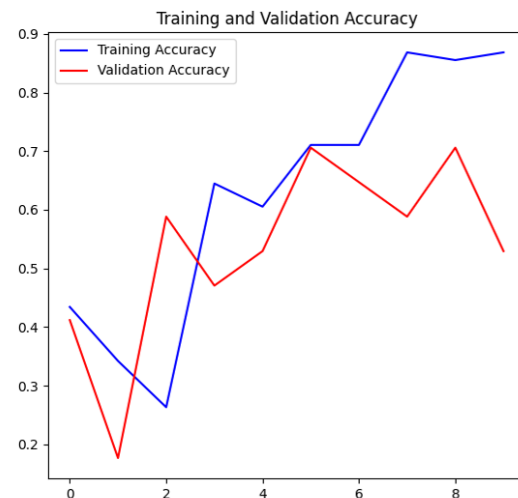
No	Hardware	Tipe Hardware
1	Processor	Ryzen 7 4700u
2	RAM	8 GB
3	Hardisk	SSD M.2 NVME SKhynix 512 GB
4	Laptop	Lenovo Ideapad Slim 3
5	Smartphone	Realme 6

4.3. Training Model CNN

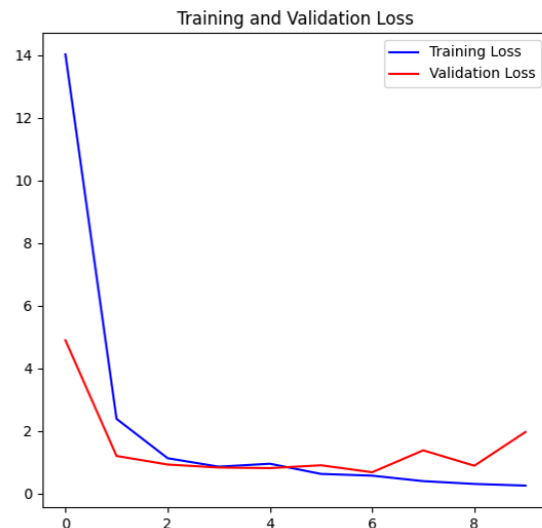
Proses diawali dengan melakukan train dan validation dari data yang sudah diambil sebelumnya menggunakan program python. Sehingga model yang dihasilkan nanti, diharapkan bisa mengklasifikasikan sayuran dengan kualitas segar dan busuk dengan baik. Setelah melakukan proses train, dilanjut untuk mengimpor modul-modul yang dibutuhkan dalam proses train ini. Mulai dari import tensorflow, matplotlib, numpy, dan CV2. Kemudian setelah proses import modul dilanjut untuk proses memuat dataset ke dalam program python. Lanjut untuk proses membuat model CNN. Semua gambar yang sudah dimuat sebelumnya akan dijadikan ke dalam ukuran yang sama, yaitu 224x224 piksel dan dengan 3 warna RGB. Kemudian model menggunakan konfigurasi train optimal adam dan matriks akurasi. Dimana nanti hasil dari proses train menggunakan metode CNN tersebut akan disimpan ke dalam model keras dengan ekstensi h5. Dibawah ini tingkat akurasi dari model yang sudah dilakukan train yang bisa dilihat pada Gambar 4 dan 5.

Gambar 4 menunjukkan akurasi dari pelatihan dan validasi. Dimana pada grafik tersebut bisa dilihat kualitas dari model yang sudah dihasilkan selama proses training berlangsung. Dimana garis biru yang menunjukkan akurasi pelatihannya meningkat seiring waktu dan pada akhirnya stabil pada angka 80%. Sementara garis merah yang ada pada gambar menunjukkan akurasi validasi model, jika dilihat pada Gambar 4 grafik sempat mengalami naik turun hingga proses terakhir. Namun akurasi validasi ini sempat menyentuh angka 70%. Grafik yang naik turun ini

tentunya bukan hal yang bagus, sehingga tentunya nantinya hal ini bisa dikembangkan lagi lebih lanjut untuk menghasilkan model yang lebih bagus.



Gambar 4. Training and Validation Accuracy



Gambar 5. Training and Validation Loss

Pada Gambar 5 menunjukkan training dan validation loss. Garis biru menunjukkan training loss, jika dilihat pada Gambar 5 garis biru menunjukkan grafik yang berkurang secara drastis hingga proses terakhir. Kemudian garis merah menunjukkan validation loss, Dimana pada gambar bisa dilihat grafik mengalami penurunan, namun ketika proses hampir selesai grafik mengalami sedikit peningkatan. Sama seperti sebelumnya hal ini tentu memerlukan beberapa proses tambahan untuk menghasilkan model yang lebih maksimal lagi.

4.4. Pengujian Model

Setelah model berhasil dilakukan train, selanjutnya model tersebut akan dilakukan pengujian untuk menguji kualitasnya. Terdapat 5 tes uji, dimana 1 jenis sayur dengan kualitas segar dan busuk akan

digunakan untuk pengujian. Tabel pengujian bisa dilihat pada Tabel 3.

Tabel 3. Tabel Pengujian Model

No	Jenis Sayur	Uji 1	Uji 2	Uji 3	Uji 4	Uji 5
1	Timun Segar	Segar	Busuk	Segar	Segar	Segar
2	Timun Busuk	Busuk	Busuk	Busuk	Busuk	Busuk
3	Selada Segar	Segar	Segar	Segar	Segar	Segar
4	Selada Busuk	Busuk	Busuk	Segar	Busuk	Busuk
5	Kubis Segar	Segar	Segar	Segar	Segar	Busuk
6	Kubis Busuk	Busuk	Segar	Busuk	Busuk	Segar
7	Wortel Segar	Segar	Segar	Segar	Segar	Segar
8	Wortel Busuk	Busuk	Busuk	Busuk	Busuk	Busuk
9	Brokoli Segar	Segar	Segar	Segar	Busuk	Segar
10	Brokoli Busuk	Busuk	Busuk	Segar	Busuk	Busuk

4.5. Evaluasi Model

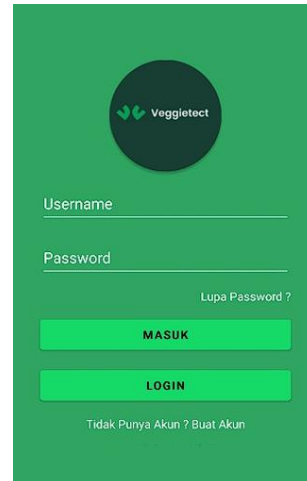
Terakhir proses mengevaluasi model yang sudah digunakan menggunakan confusion matrix. Pada confusion matrix ini nantinya akan menampilkan persentase sukses dan gagal dari model yang sudah dibuat. Dimana nantinya akan menghasilkan informasi mengenai label actual (sebenarnya) dan label prediksi (predicted). Confusion matrix ini juga bisa digunakan untuk menghitung akurasi, presisi, recall, dan skor F1 dari model. Tabel confusion matrix bisa dilihat pada Tabel 4.

Tabel 4. Tabel Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	22 (TP)	3 (FN)
Actual Negative	4 (FP)	21 (TN)

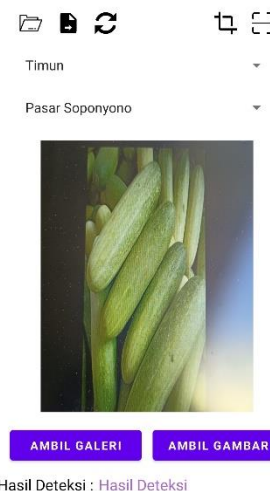
4.6. Implementasi Aplikasi

Pertama-tama ketika pengguna pertama kali membuka aplikasi akan terbukan menu halaman login. Dimana pengguna perlu melakukan login terlebih dahulu sebelum mengakses aplikasi. Pengguna yang tidak mempunyai akun bisa mendaftar pada menu buat akun. Jika sudah selesai mendaftar pengguna bisa memasukkan username dan password yang sudah dibuat sebelumnya. Jika tidak cocok dan sesuai akan muncul notif peringatan. Tampilan halaman login bisa dilihat pada Gambar 6.



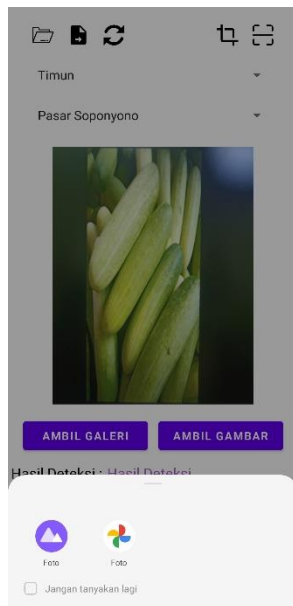
Gambar 6. Tampilan Halaman Login

Setelah pengguna berhasil melakukan login, akan masuk ke dalam tampilan utama aplikasi. Pada menu ini pengguna bisa melakukan deteksi sayur. Awalnya pengguna memilih jenis sayuran yang ingin dideteksi. Kemudian bisa memilih dari pasar mana sayur tersebut diambil. Proses ini bisa dilihat pada Gambar 7.



Gambar 7. Tampilan Menu Utama

Jika pengguna memilih untuk mengambil gambar dari galeri akan muncul pilihan melalui aplikasi yang diinstal pengguna. Namun jika ingin mengambil gambar secara langsung dari aplikasi bisa menekan tombol ambil gambar untuk mendapatkan gambar. Sehingga ketika ditekan gambar akan otomatis langsung tampil di halaman. Hal ini bisa dilihat pada Gambar 8.

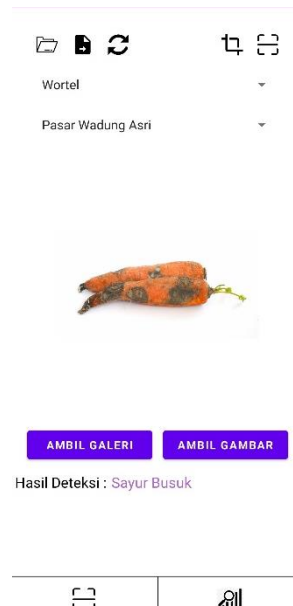


Gambar 8. Tampilan Memilih Gambar dari Galeri

Apabila gambar sudah diambil pengguna bisa lanjut untuk mengklasifikasikan sayuran tersebut. Apabila gambar kurang jelas, pengguna bisa memotong gambar pada bagian sayurnya dengan menggunakan tombol crop. Jika dirasa sudah jelas pengguna bisa menekan tombol scan untuk mengklasifikasikan sayur. Apakah sayur tersebut termasuk sayur segar atau sayur busuk. Setelah hasil klasifikasi sayur muncul, bisa lanjut untuk mengirim data ke server untuk analisis tiap pasar nantinya. Hal ini bisa dilihat pada Gambar 9 dan 10.



Gambar 9. Tampilan Deteksi Sayur Segar



Gambar 10. Tampilan Deteksi Sayur Segar dan Busuk

Selain bisa mengklasifikasikan sayur, aplikasi juga bisa melihat analisis grafik dari tiap pasar yang sudah diambil data sayurnya. Sebelumnya peneliti bisa mengunjungi 3 pasar yang ada di daerah Surabaya. Sehingga pada aplikasi terdapat 3 pilihan menu pasar yang bisa dilihat analisis grafiknya. Setelah pengguna memilih salah satu pasar yang ingin dilihat analisisnya, akan muncul grafik yang memperlihatkan persentase sayur segar dalam satu bulan terakhir. Sehingga pengguna bisa melihat apakah pada pasar tersebut mempunyai persentase sayur segar yang baik. Hal ini bisa memudahkan pengguna untuk membeli sayur yang memiliki kualitas bagus.

4.7. Pengujian Akurasi

Berdasarkan hasil pengujian model pada tabel 3 terdapat total 50 tes pengujian yang dilakukan pada 5 jenis sayur segar dan busuk. Dari 50 tes pengujian didapat 22 prediksi benar untuk kelas “Segar” (TP) dan 21 prediksi benar untuk kelas “Busuk” (TN). Kemudian ada 3 kelas “Segar” salah diprediksi sebagai kelas “Busuk” (FN) dan 4 kelas “Busuk” salah diprediksi sebagai kelas “Segar” (FP). Dari hasil pengujian tersebut bisa dihitung akurasi dari model:

$$\text{Akurasi} = (\text{TP} + \text{TN}) / \text{Total Pengujian} \times 100\% \quad (1)$$

$$\text{Akurasi} = (22 + 21) / 50 \times 100\% = 86\%$$

Jadi, Tingkat akurasi dari model yang dihasilkan mencapai nilai 86%.

5. KESIMPULAN DAN SARAN

Dari hasil pengujian yang sudah dilakukan dapat dibuat sebuah kesimpulan dengan menggunakan metode CNN dalam mengklasifikasikan sayur segar dan busuk memberikan hasil yang baik. Dimana berdasarkan tes pengujian dari model yang dihasilkan dengan menggunakan metode CNN mempunyai akurasi lumayan bagus. Dari sekitar 10 jenis sayuran

dan 5 tahap pengujian, sehingga ada sekitar 50 tes yang dijalankan. Dari 50 tes yang dijalankan, terdapat sekitar 7 kesalahan dalam pengujian. Jadi akurasi dari model yang dihasilkan memiliki nilai 86%.

Dengan hasil akurasi yang lumayan bagus diatas 80%, dimana hasil tes ini diperoleh dengan menggunakan optimizer Adam sebagai konfigurasi model. Sehingga dengan tingkat akurasi yang cukup tinggi ini diharapkan penelitian ini bisa membantu masyarakat untuk memilih sayuran yang segar dan berkualitas. Tidak hanya itu, dengan adanya deteksi sayur ini diharapkan minat Masyarakat dalam mengkonsumsi sayuran bisa meningkat. Sehingga penelitian ini juga bisa membantu masyarakat yang mulai mengkonsumsi sayuran untuk bisa memilih sayur layak konsumsi. Namun penelitian ini tentu memiliki kekurangan dan kelebihan juga.

Berdasarkan penelitian yang sudah dilakukan, ada banyak kekurangan didalamnya, sehingga untuk menutupi berbagai kekurangan yang ada di penelitian ini, ada beberapa saran yang bisa digunakan untuk mengembangkan lagi jumlah datasetnya mulai dari kualitas dan kuantitas gambar yang diambil. Sehingga model yang dihasilkan nanti bisa lebih maksimal dan lebih bagus lagi akurasinya. Selain menambah dataset, tentunya fitur yang digunakan untuk melatih model bisa ditambahkan selain dari bentuk, warna, dan batangnya. Sehingga bisa menghasilkan model yang lebih bagus lagi dalam mengidentifikasi sayur. Penelitian berikutnya juga bisa menambahkan beberapa kategori dalam deteksi sayurannya, mulai dari menambahkan Tingkat persen segar dan busuk sayur, saran untuk menyimpan sayur, rekomendasi masakan yang bisa dibuat. Penelitian lebih lanjut dapat diarahkan ke deteksi dini penyakit tanaman, memungkinkan petani untuk mengambil tindakan pencegahan yang lebih cepat dan efektif.

DAFTAR PUSTAKA

- [1] A. Wulansari, A. W. Sari, dan A. Ninda, "Edukasi Mitos dan Fakta Sayur dan Buah Bagi Remaja di SMA N 9 Tanjung Jabung Timur," *J. Abdimas Kesehat.*, vol. 5, no. 3, hal. 569, 2023, doi: 10.36565/jak.v5i3.583.
- [2] D. T. Yulianto, F. Juita, E. H. Yulianto, A. F. Pertanian, dan U. Mulawarman, "(Studi Kasus Pasar Segiri Kota Samarinda)," hal. 1489–1498, 2024.
- [3] M. Setiono dan Supatman, "Klasifikasi Penyakit Antraknosa Citra Cabai Rawit Dengan Metode Convolutional Neural Network (CNN)," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 11, no. 2, hal. 308–320, 2024.
- [4] P. W. R. Pamungkas dan R. Khalida, "Manajemen Proyek Agile dengan Pendekatan Metode Scrum sebagai Peningkatan Layanan Berkelanjutan Perusahaan," *Sisfotek*, vol. 3, no. 1, hal. 187–194, 2019.
- [5] M. R. S. Alfarizi, M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, dan M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karya Ilm. Mhs. Bertauhid (KARIMAH TAUHID)*, vol. 2, no. 1, hal. 1–6, 2023.
- [6] Edwin Febrywinata, "Pengenalan Dan Klasifikasi Jenis Buah Menggunakan Metode CNN Secara Sederhana Dengan Menggunakan Google Colab," *Merkurius J. Ris. Sist. Inf. dan Tek. Inform.*, vol. 2, no. 4, hal. 185–193, 2024, doi: 10.61132/mercurius.v2i4.162.
- [7] M. I. Mustofa, A. Larasati, R. Febrian, dan S. Komariyah, "Perancangan Mobile App Food Oder Master Seafood," *INSTINK (Jurnal Inov. Pendidikan, Teknol. Inf. Komput.)*, vol. 3, no. 1, hal. 41–50, 2024.
- [8] A. P. A. Gita, N. T. Surya, dan A. Setyaningsih, "Aplikasi stunting berbasis android guna mempercepat deteksi dini kejadian stunting," *J. Public Heal. Innov.*, vol. 3, no. 02, hal. 142–150, 2023, doi: 10.34305/jphi.v3i02.714.
- [9] L. Fadhillah, W. Hadikurniawati, dan U. Semarang, "1,2 1 , 2," vol. 7, 2024.
- [10] A. R. Hidayat dan V. Lusiana, "Deteksi Jenis Sayuran dengan Tensorflow Dengan Metode Convolutional Neural Network," *J. Sains Komput. Inform. (J-SAKTI)*, vol. 6, no. 2, hal. 1032–1040, 2022.
- [11] R. D. Rusdiyan Yusron dan M. M. Huda, "Analisis Perancangan Sistem Informasi Perpustakaan Menggunakan Model Waterfall Dalam Peningkatan Inovasi Teknologi," *J. Autom. Comput. Inf. Syst.*, vol. 1, no. 1, hal. 26–36, 2021, doi: 10.47134/jacis.v1i1.4.
- [12] M. R. F. Hadi, "Natural Language Processing menggunakan Metode Fuzzy String Matching pada Chatbot berbasis Web pada Aplikasi Whatsapp," 2022, [Daring]. Tersedia pada: http://digilib.uinsby.ac.id/id/eprint/51730%0Ahttp://digilib.uinsby.ac.id/51730/2/MuhammadRifqyFakhrulHadi_H06217015.pdf
- [13] A. Novia Ramadani, Z. Muzyaitir, dan U. Negeri Makassar, "Journal of Renewable Energy and Smart Device PENGEMBANGAN SISTEM INFORMASI KEUANGAN SEKOLAH MENGGUNAKAN METODE WATERFALL," hal. 31–36, 2023.
- [14] M. Yulianti, U. Pamulang, dan T. Selatan, "Sistem Informasi Pendaftaran Peserta Didik Baru (Ppdb) Smk Iptek Tangsel Berbasis Web," vol. 1, no. 3, hal. 485–495, 2023.