

KLASIFIKASI JENIS JAMBU BIJI BERDASARKAN TEKSTUR DAUN MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORKS (CNN)

Kamil Malik

Fakultas Teknik Universitas Nurul Jadid

Paiton Probolinggo 67291

e-mail: kamil@unuja.ac.id

ABSTRAK

Jambu biji adalah tanaman buah yang banyak berkembang di daerah dengan iklim tropis ataupun subtropis, seperti di negara Mexico, Amerika Selatan, Indonesia dan negara disekitarnya. Beberapa jenis jambu memiliki daun yang mirip, maka akan sulit untuk membedakan jenisnya. Salah satu alternatif yang dapat digunakan untuk permasalahan tersebut adalah klasifikasi menggunakan teknik *Image Processing* (pengolahan citra) dengan menggunakan metode *Convolutional Neural Networks (CNN)*. Pengolahan citra pada daun jambu dilakukan untuk mengenali warna dan teksturnya. selanjutnya akan di proses menggunakan metode *Convolutional Neural Networks (CNN)*. Jumlah Dataset yang digunakan sebanyak 880 gambar, 640 sebagai data train, 160 sebagai data validasi dan 80 sebagai data uji, data tersebut dibagi menjadi 4 kelas. Hasil dari proses pelatihan terhadap model pada penelitian ini mendapat nilai akurasi sebesar 97%. Sedangkan pada pengujian terhadap data baru menggunakan data test pada dataset dimana model di uji menggunakan 20 data citra daun jambu pada masing-masing kelas dan mendapatkan hasil rata-rata nilai akurasi sebesar 93%.

Kata Kunci: Jambu biji, Image Processing, Klasifikasi, CNN

ABSTRACT

Guava is a fruit that is widely grown in areas with tropical or subtropical climates, such as in Mexico, South America, Indonesia and surrounding countries. Some types of guava have similar leaves, so it will be difficult to distinguish the type. An alternative that can be used is classification using Image Processing, namely the *Convolutional Neural Networks (CNN)* method. Image processing on guava leaves is done to recognize the color and texture. then it will be processed using the *Convolutional Neural Networks (CNN)* method. The dataset used is 880 images. 640 as train data, 160 as validation data and 80 as test data, the data is divided into 4 classes. The final accuracy result is 97%. Testing on 20 new datasets resulted in an accuracy of 93%.

Keywords: Guava, Image Processing, classification, CNN

I. PENDAHULUAN

Tanaman jambu biji merupakan buah yang berasal dari daerah tropis maupun subtropic.[1] Jenis tanaman jambu biji ini beranekaragam diantaranya adalah jambu Australia, jambu gunung, jambu kristal, jambu merah dan jambu mini.[2] Tanaman jambu biji jika sudah tampak buahnya akan mudah sekali dibedakan jenisnya terutama saat buah telah siap untuk dikonsumsi. Disisi lain, tanaman jambu biji jika masih dalam kondisi belum berbuah atau hanya tampak pohon dan daunnya saja akan sulit dibedakan dikarenakan Daun jambu biji bervariasi dalam bentuk dan ukuran serta menunjukkan keragaman genetik yang cukup luas.[3] Oleh karenanya, tujuan dari penelitian ini adalah untuk mengetahui jenis tanaman jambu biji berdasarkan tekstur daunnya.

Kesulitan membedakan jenis tanaman jambu biji ini sering menjadi permasalahan di dunia pertanian, terutama bagi petani pemula. Terkadang mereka terjebak dalam pemilihan jenis tanaman jambu biji ini yang mengakibatkan saat panen tidak sesuai harapan. Dengan penelitian ini harapannya adalah program yang dihasilkan dapat membantu petani dalam mendeteksi dini jenis tanaman jambu biji berdasarkan tekstur

daunnya, sehingga petani bisa mengetahui jenis tanaman jambu biji apa yang akan mereka tanam.

Daun tanaman merupakan salah satu biometric dan karakteristik tanaman. Hal ini dikarenakan bentuk daun dan tulang daun berbeda-beda tergantung dari jenis tanamannya. Struktur tulang daun merupakan salah satu ciri khas daun yang berperan penting dalam mengklasifikasikan jenis tumbuhan melalui serangkaian proses pencitraan [3]

Timbul permasalahan ketika seseorang melihat tanaman jambu dan ingin mengetahui jenis jambu yang dilihatnya. Beberapa jenis jambu memiliki daun yang mirip, maka akan sulit untuk membedakan jenisnya. Untuk mengatasi keterbatasan manusia untuk mengetahui jenis jambu maka diperlukan adanya penelitian sebagai solusi. Penelitian ini diharapkan dapat membantu mengklasifikasi jenis jambu dari gambar daun yang dimasukkan dan diproses menggunakan metode. Sehingga dibutuhkan metode yang dapat menggolongkan jambu secara otomatis melalui serangkaian proses pengolahan citra daun jambu.

Salah satu alternatif yang dapat digunakan untuk permasalahan tersebut adalah menggunakan teknik *Image Processing* (pengolahan citra) dengan menggunakan metode *Convolutional Neural Networks (CNN)*. Pengolahan citra pada daun jambu dilakukan untuk memproses citra daun jambu dengan

mengenali warna dan teksturnya selanjutnya hasil akan di proses lebih lanjut menggunakan metode Convolutional Neural Networks (CNN). Pada penelitian ini metode Convolutional Neural Networks (CNN) di gunakan sebagai salah satu cara untuk membantu proses pengklasifikasian objek.

Berdasarkan latar belakang dari permasalahan tersebut, judul skripsi yang diambil pada penelitian ini adalah “Klasifikasi Jenis Jambu Berdasarkan Warna Dan Tekstur Daun Menggunakan Metode Convolutional Neural Network (CNN). Dengan penelitian ini diharapkan proses klasifikasi jenis jambu dapat membantu dalam pengenalan jambu dari daun untuk menentukan jenis jambu yang dilakukan dengan mudah, cepat, dan akurat.

II. LANDASAN TEORI

A. Jambu

Jambu biji atau dalam Bahasa ilmiahnya disebut *Psidium guajava* L. dan lebih populer disebut *guava*, adalah jenis tanaman suku *myrtle* (*Myrtaceae*). Kandungan pada buah dan daun jambu biji sangat banyak, sehingga sering dijadikan sebagai bahan untuk obat tradisional.[1] Salah satu penamfaatan daun biji adalah untuk obat diare[5] sedangkan buahnya sering digunakan untuk suplemen makanan karena banyak mengandung vitamin C.[6]

Buah jambu biji (*Psidium guajava*) atau Guava merupakan salah satu buah yang bisa tumbuh di daerah beriklim tropis maupun subtropis, seperti di Mexico, Amerika Selatan, Indonesia maupun negara-negara disekitarnya.[1]

Manfaat jambu biji tidak hanya terdapat pada buahnya tapi juga pada daunnya. Daun jambu biji mengandung senyawa aktif tanin, triterpenoid, flavonoid, dan saponin. Tujuan penelitian untuk mengetahui perbedaan efektivitas dan daya hambat dari konsentrasi 60%, 50%, 40%, 30%, 20%, dan 10% antara ekstrak daun jambu biji buah merah dan putih terhadap pertumbuhan *Lactobacillus acidophilus* ATCC® 4356TM.[2]

Daun pada tanaman jambu biji memiliki struktur daun tunggal dan mengeluarkan aroma yang khas jika diremas. Kedudukan daunnya bersilangan dengan letak daun berhadapan dan pertulangan daun menyirip. Terdapat beberapa bentuk daun pada tanaman jambu biji, yaitu: bentuk daun lonjong, jorong, dan bundar telur terbalik. Bentuk daun yang paling dominan adalah bentuk daun lonjong. Perbedaan pada bentuk daun dapat dipengaruhi oleh faktor genetik dan faktor lingkungan.[7] Daun jambu biji umumnya berwarna hijau pada tiap varietas, berbeda pada Varietas Australia yang memiliki warna daun merah. Jambu biji Varietas Australia memiliki ciri khas yang unik, yaitu batang, daun, bunga, maupun buahnya berwarna merah tua[7].

Tanaman jambu biji (*Psidium guajava*) berdaun tunggal dengan bentuk lonjong, tepi tumpul, bagian bawah membulat, dan tepi rata Daun jambu biji memiliki panjang 6-14 cm dan lebar 3-6 cm. Daun ini memiliki warna hijau kekuning-kuningan dan memiliki daun berduri Panjang. Daunnya berbentuk lonjong, dengan tepi atas dan bawah melengkung ke atas. Buahnya bulat sampai lonjong, merah, hijau sampai kuning-hijau. Buah pohon ini sangat lebat dan terdapat di tengah pohon, dengan biji yang banyak[3].

Untuk melihat ciri-ciri daun jambu kristal yakni terletak pada bentuk daun yang terlihat lebih lebar dan memiliki ujung daun yang lebih bulat membulat serta tekstur daun yang

terlihat lebih halus. Gambar 2.1 menunjukkan gambar daun jambu kristal.



Gambar 1. Daun Jambu Kristal

Untuk melihat ciri-ciri daun jambu gunung yakni terletak pada bentuk daun yang terlihat lebih lonjong dan memiliki ujung daun yang sedikit lancip serta tekstur daun yang lebih kasar. Gambar 2 menunjukkan gambar daun jambu gunung.



Gambar 2 Daun Jambu Gunung

Untuk melihat ciri-ciri daun jambu merah australia yakni terletak pada bentuk daun yang terlihat lebih kecil dan lonjong serta memiliki ujung daun yang sedikit runcing serta tekstur daun yang terlihat lebih halus dan warna daun yang berwarna merah. Gambar 3 menunjukkan gambar daun jambu merah australia.



Gambar 3 Daun Jambu Merah Australia

Untuk melihat ciri-ciri daun jambu mini yakni terletak pada bentuk daun yang terlihat lebih kecil dan memiliki ujung daun yang runcing serta tekstur daun yang terlihat lebih kasar serta warna daun yang sedikit lebih gelap. Gambar 4 menunjukkan gambar daun jambu mini.



Gambar 4 Daun Jambu Mini

B. Google Colab

Google Colaboratory atau yang lebih umum dengan sebutan *google colab*, adalah sebuah layanan *free cloud* yang disediakan oleh google dalam bentuk *executable document* yang dapat digunakan untuk menyimpan, menulis, mengedit serta membagikan program yang telah disimpan pada *Google*

Klasifikasi Jenis Jambu Biji berdasarkan tekstur daun menggunakan *Convolutional Neural Networks (CNN)*

Drive. Software colab ini hampir mirip dengan *Jupyter Notebook*, sebuah cloud yang dijalankan dengan perantara browser, seperti Google Chrome dan Mozilla Firefox[8].

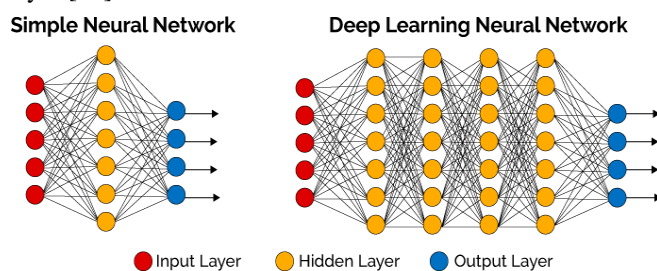
Colab dapat mengimpor set data gambar, melatih pengklasifikasi gambar di dalamnya, dan mengevaluasi model, semuanya hanya dalam beberapa baris kode. *Notebook Colab* mengeksekusi kode di *server cloud* Google, artinya dapat memanfaatkan kekuatan perangkat keras Google, termasuk GPU dan TPU, terlepas dari kekuatan mesin [9].

C. Klasifikasi

Klasifikasi merupakan suatu bentuk analisa data yang memperoleh pola-pola yang menggambarkan kategori-kategori penting dari data. Klasifikasi ini melibatkan dua tahap proses. Fase pertama disebut juga fase pembelajaran (pengembangan model klasifikasi) dan fase kedua adalah fase klasifikasi, dimana model digunakan untuk memprediksi skor class dari data yang diberikan. Keakuratan model klasifikasi bisa ditentukan oleh beberapa percobaan yang diberikan dengan menghitung persentase data test yang diklasifikasikannya dengan akurat oleh model classification. Apabila akurasi *classification* dianggap baik, berarti *classification* tersebut bisa dipergunakan sebagai *classification* data lain[10]

D. Deep Learning

Deep Learning merupakan cabang ilmu pembelajaran untuk dirancang melakukan suatu analisa data dan struktur logika. Deep Learning memiliki tool-nya sendiri yang didalamnya terdapat sebuah struktur algoritma yang berlapis – lapis atau yang lebih dikenal dengan sebutan jaringan saraf tiruan. Deep Learning dapat mengetahui dan mampu melakukan klasifikasi suatu objek, suara, dan text. Pembelajaran mendalam ini merupakan bagian dari kecerdasan buatan (AI) dan pembelajaran mesin (ML). Deep learning juga hasil dari fork machine learning yang mengimplementasikan jaringan syaraf tiruan yang secara alami memiliki banyak hidden layer.[11]



gambar 5: Lapisan Layer Deep Learning
Sumber: [12]

Gambar 1 di atas merupakan contoh lapisan pada layer deep learning. Lingkaran yang berwarna merah yaitu lapisan (layer input), lingkaran berwarna kuning yaitu lapisan tersembunyi (layer hidden), lingkaran berwarna merah yaitu lapisan keluaran (output layer) dan lingkaran-lingkaran tersebut merepresentasikan neuron. Pada setiap (hidden layer) atau lapisan tersembunyi mempunyai beberapa atau lebih dari satu neuron.

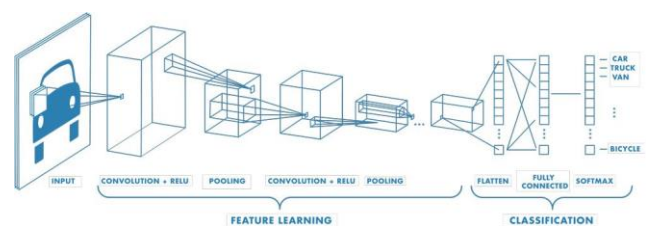
E. *Convolutional Neural Networks (CNN)*

Convolutional Neural Network adalah algoritma pembelajaran mendalam yang merupakan peningkatan persepsi

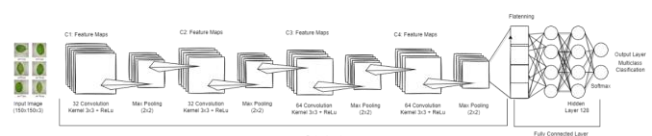
multilayer (MLP) yang dirancang untuk memproses data dalam format 2 dimensi, seperti gambar dan suara. CNN digunakan untuk mengklasifikasikan data berlabel menggunakan teknik pembelajaran yang diawasi (Supervised Learning). Supervised Learning mempunyai sistem data yang sudah dilatih dan ada variabel target, jadi metode ini bertujuan untuk mengelompokkan data, dengan mengolah data yang ada.

Convolutional Neural Network (CNN) memiliki lapisan rangkaian neuron tiga dimensi (tinggi, lebar dan kedalaman). Lebar dan tinggi mewakili ukuran lapisan, dan kedalaman menunjukkan jumlahnya lapisan. Convolutional Neural Network (CNN) bisa mempunyai puluhan sampai ratusan lapisan, yang masing-masing belajar untuk mengenali gambar yang berbeda. Pemrosesan citra diterapkan pada setiap data training dengan ukuran resolusi berbeda, dan keluaran dari setiap gambar yang diproses dan dipakai untuk proses selanjutnya. Pemrosesan gambar dimulai dari fitur sederhana seperti cahaya dan sudut, atau membuat fitur kompleks yang mengidentifikasi objek unik berdasarkan ketebalan lapisan.[13]

Pada gambar 2, Convolutional Neural Network (CNN) dibagi menjadi dua lapisan. Lapisan pertama merupakan lapisan khusus yang terletak di awal arsitektur, yang memiliki beberapa lapisan dan masing-masing memiliki neuron yang terhubung ke wilayah lapisan sebelumnya. Jenis lapisan pertama merupakan lapisan konvolusi dan lapisan berikutnya merupakan pooling layer(pengelompokan). Setiap level diberikan fungsi aktivasi yang beralih antara mode pertama dan kedua. Lapisan ini menerima dan dengan cepat memproses gambar input untuk membuat gambar vektor yang diproses oleh lapisan selanjutnya. Lapisan kedua merupakan lapisan classification layer setiap lapisan mengandung neuron yang terhubung dengan lapisan lainnya. Lapisan ini mengambil input dari lapisan ekstraksi fitur gambar sebagai vektor dan kemudian mengubahnya menjadi jaringan Multi-Neural dengan penambahan beberapa lapisan tersembunyi (hidden layer). Outputnya adalah akurasi kelas untuk klasifikasi. [14]



Gambar 6 Arsitektur CNN secara umum
Sumber: [13]

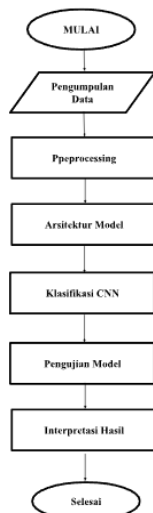


Gambar 7 Simulasi Arsitektur CNN pada daun Jambu

III. METODOLOGI PENELITIAN

Penelitian ini menggunakan metode kuantitatif karena data pada penelitian ini berupa angka-angka dan analisis menggunakan statistik penelitian yang menekankan pada analisis data-data numerik (angka) yang diolah dengan metode statistik. Pada dasarnya, pendekatan kuantitatif dilakukan pada penelitian inferensial (pengujian hipotesis) dan menyandarkan kesimpulan hasilnya pada suatu probabilitas kesalahan penolakan hipotesis nol (nihil). Dengan metode kuantitatif, diperoleh signifikansi perbedaan kelompok atau hubungan antar variable yang jelas[15].

Kerangka atau perancangan sistem yang akan digunakan atau dibangun dalam penelitian ini meliputi beberapa tahap yakni pengumpulan dataset, preprosesing, arsitektur model, pengklasifikasian dengan menggunakan *Deep Learning* dan metode *Convolution Neural Network (CNN)*, pengujian model dan interpretasi hasil. Dimulai dari pengumpulan data yang di lakukan dengan cara mengumpulkan dataset daun jambu sebanyak 750 gambar dataset, lalu di lanjut dengan tahapan preprocessing (pra pemrosesan) yang berguna untuk menyiapkan berbagai data citra daun jambu agar digunakan dengan optimal, selanjutnya dilakukan tahapan arsitektur model yang digunakan untuk pemodelan dengan api keras yang terdapat dari beberapa fungsi untuk memanggil pembuatan model yakni model sequental, dilanjutkan dengan tahapan selanjutnya yakni dengan tahapan Klasifikasi yakni tahap pengklasifikasian data citra daun jambu yang telah di normalisasikan lalu di lakukan proses klasifikasi dengan algoritma CNN pada lapisan terakhir model, citra yang di input selanjutnya akan di proses melalui konvolusi yang berdasarkan pada lapisan konvolusi pada model CNN, selanjutnya di lakukan tahapan pengujian model yakni proses yang di lakukan setelah mendapatkan model dari proses training oleh CNN untuk mengevaluasi model yang di buat agar supaya dapat dikenali datasetnya, proses pengujian model ini merupakan proses memasukkan file citra daun jambu dari data testing yang sudah tersedia pada dataset, dan di tahapan terakhir yakni implementasi hasil yakni pengujian model yang di lakukan setelah mendapatkan model dari proses training oleh CNN dengan model yang nantinya akan menghasilkan sebuah output dan di hitung dari bobot yang telah ditingkatkan. Kemudian akan dibandingkan dengan bobot model dari data train, sehingga model klasifikasi data train dapat mengklasifikasikan data uji tersebut ke label atau kelas yang sesuai.



Gambar 8. Metode penelitian

Gambar 8 di atas dapat dijabarkan sebagai berikut:

1. Pengumpulan data

Pengumpulan data dilakukan dengan cara mengumpulkan dataset daun jambu. Dalam dataset tersebut, terdapat 750 data gambar daun jambu dengan total ukuran 305 MB dan ukuran pixel dataset yang masih bawaan (original) atau belum dinormalisasikan Ada lima jenis dataset yang nantinya akan digunakan pada penelitian ini yaitu jambu Australia, jambu gunung, jambu kristal, jambu merah dan jambu mini.

2. Preprocessing

Tahapan preprocessing (pra-pemrosesan) yang berguna untuk menyiapkan berbagai data citra daun jambu agar dapat digunakan dengan Optimal. Yaitu dengan cara me-*resize* semua ukuran citra agar semuanya sama. Karena citra daun jambu dari dataset yang telah dikumpulkan memiliki banyak ukuran piksel yang berbeda-beda.

3. Arsitektur Model CNN

Arsitektur model yang digunakan merupakan model MLP dengan banyak lapisan. Pada API Keras terdapat beberapa fungsi untuk memanggil pembuatan model yakni model Sequential. Dengan model Sequential cukup mudah untuk membuat beberapa jaringan syaraf dengan banyak lapisan

4. Klasifikasi CNN

Data citra daun jambu yang telah dinormalisasi berikutnya sudah siap untuk dilakukan proses klasifikasi dengan algoritma CNN. Proses klasifikasi dilakukan pada lapisan terakhir model. Citra input selanjutnya akan diproses melalui konvolusi dan proses pooling pada tahapan feature learning. Jumlah proses konvolusi yang berdasarkan lapisan konvolusi pada model CNN. Jumlah filter dan ukuran kernel memiliki beberapa ukuran berbeda-beda pada setiap proses konvolusi. Kemudian dilakukan proses flatten atau proses mengubah feature map hasil pooling layer ke dalam bentuk vektor. Proses ini biasa juga disebut dengan tahap fully Connected layer.

5. Pengujian Model

Pengujian model dilakukan dengan penginputan file citra daun jambu dari data testing yang ada pada dataset kedalam model fit, yang selanjutnya akan menghasilkan output dan menghitung bobot tambahan. Jika berhasil maka dilanjutkan ke proses penghitungan akurasi sedangkan jika tidak berhasil maka kembali ke proses klasifikasi cnn. Setelah melakukan pengujian data maka selanjutnya melakukan pengujian evaluasi performa menggunakan *confusion matrix* dan *classification report* untuk mengukur performa model klasifikasi berupa *akurasi*, *precision*, dan *recall*. Formulanya Nampak pada rumus nomor 1,2 dan 3.

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

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (3)$$

Keterangan :

TP (True Positif) = Jumlah data true terdeteksi positif
 TN (True Negative) = Jumlah data true terdeteksi negative
 FP (False Positif) = Jumlah data false terdeteksi positif
 FN (False Negative) = Jumlah data false terdeteksi negative

6. Implementasi Hasil

Klasifikasi Jenis Jambu Biji berdasarkan tekstur daun menggunakan *Convolutional Neural Networks (CNN)*

Pengujian model akan dilakukan setelah mendapatkan model dari proses training oleh CNN untuk mengevaluasi model yang dibuat agar dapat mengenali datasetnya. Proses pengujian model ini merupakan proses memasukkan file citra daun jambu dari data testing yang sudah tersedia pada dataset ke dalam `fit()` model yang nantinya menghasilkan sebuah output dan dihitung bobot yang telah ditingkatkan. Kemudian akan dibandingkan dengan bobot model dari data train, sehingga model klasifikasi data train dapat mengklasifikasikan data uji tersebut ke label atau kelas yang sesuai.

Hasil akurasi merupakan salah satu indikator untuk mengevaluasi model klasifikasi. Dimana sistem klasifikasi akan memperoleh hasil output berupa akurasi dari proses pengujian model. Untuk melakukan pengujianya dengan menggunakan rumus yang mampu menghitung nilai akurasi dengan menggunakan rumus 4.

$$Akurasi = \frac{\sum \text{Data benar}}{\sum \text{Data uji}} \times 100\% \quad (4)$$

Keterangan :

$\sum \text{Data benar}$ = Jumlah banyaknya data yang telah teridentifikasi benar

$\sum \text{Data uji}$ = Jumlah banyaknya data keseluruhan yang dipakai

IV. HASIL DAN PEMBAHASAN

1. Pengumpulan Dataset

Dataset yang digunakan pada penelitian ini adalah citra daun jambu gunung, jambu merah australia, jambu kristal dan jambu mini. Jumlah gambar yang ada pada dataset adalah 1.100 sampel gambar yang terbagi menjadi 800 sampel data training dan 200 sampel data validation dan 100 sampel data uji dengan resolusi yang sama yakni 150x150.

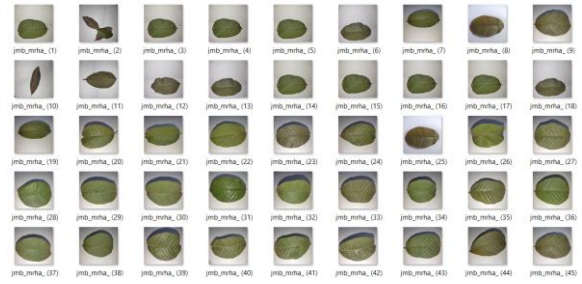
Dataset diperoleh melalui observasi atau pengamatan secara langsung yang bertujuan untuk mengetahui bagaimana karakteristik jenis jambu berdasarkan daun. Observasi dilakukan dengan cara mengamati dan memahami tekstur dan warna pada daun jambu.

Dataset dipersiapkan untuk dipelajari oleh model, dataset asli yang telah di kumpulkan kemudian akan dilakukan tahap preprocessing. Dalam preprocessing dataset tersebut, dataset akan dipelajari dari segi tampilan gambarnya, ukuran gambarnya, ukuran shape gambar dan proporsi masing-masing kelas, tujuannya yaitu untuk mendapatkan insight terhadap dataset, dataset yang di dapatkan merupakan hasil gambar asli yang di ambil secara langsung melalui kamera smartphone samsung dengan kamera 48 megapixel resolusi 9:16 dengan pengambilan gambar kurang lebih dengan jarak 5 - 8cm dengan latar belakang warna putih yang di lakukan di luar rumah pada siang hari dengan objek dataset daun jambu dengan cahaya yang maksimal sebagus mungkin dan tanpa bayangan.



Gambar 9. Proses pengambilan dataset

Pada gambar 10, gambar 11, gambar 12 dan gambar 13 merupakan gambar contoh citra dari lima jenis daun jambu.



Gambar 10. Daun Jambu Australia



Gambar 11. Daun Jambu Gunung



Gambar 12. Daun jambu kristal

2. Preprocessing

Preprocessing adalah sebuah proses yang dilakukan pada suatu citra digital sebelum sebelum dilakukan tahap citra selanjutnya. Tahapan pre-processing dibagi menjadi dua bagian diantaranya *image generator* dan *image augmentation*.

a. Image Generator

Pada sub bab ini dataset akan dinormalisasikan agar pengujian berjalan dengan optimal dengan melakukan input beberapa features seperti *rotation range*, *width shift range*, *height shift range*, *shear range*, *zoom range*, *horizontal flip*, *fill mode*, *vertical flip*, *fill mode*.

Image Generator dilakukan dalam delapan proses yaitu, Proses pertama yang dilakukan adalah import data generator dengan menggunakan bantuan keras library tensorflow. Kemudian proses kedua adalah dilakukan split train data generator dengan beberapa parameter. ketiga, gambar dirotasi sebesar 40 derajat serta dilakukan pergeseran gambar dengan menentukan batas pecahan dari total lebar dan digeser ke kiri atau ke kanan secara acak. Keempat adalah Proses peninggian gambar juga dilakukan secara acak, dengan menentukan batas pecahan dari total tinggi dan digeser ke atas atau ke bawah secara acak. Kelima gambar mengalami peregang-an atau pergeseran sudut sebesar 0.2 serta gambar diperbesar 0.1. keenam, pengisian pixel yang kosong pada gambar. Ketujuh, dilakukan memutar balik gambar

secara horizontal dan secara vertical. Langkah terakhir yaitu kedelapan dilakukan normalisasi pixel gambar antara 1-255.

Tahap selanjutnya setelah dilakukannya proses konfigurasi parameter pada *image augmentation* tahap selanjutnya adalah menentukan ukuran *batch size*, ukuran target gambar yang akan dihasilkan dan membagi gambar yang akan di *generator* ke data *train* dan data validasi. Pada penelitian ini digunakan ukuran gambar sebesar 150 x 150.

```
1. from tensorflow.keras.preprocessing.im-
   age import ImageDataGenerator
2. train_datagen = ImageDataGenerator(
3.     rotation_range=40,
4.     width_shift_range=2,
5.     height_shift_range=0.2,
6.     shear_range=0.2,
7.     zoom_range=0.1,
8.     fill_mode="nearest",
9.     horizontal_flip=True,
10.    vertical_flip=True,
11.    rescale=1./255,
```

b. Image Augmentation

Setelah melakukan proses *image generator* maka tahapan selanjutnya adalah melakukan tahapan *image augmentation* yang bertujuan untuk merubah ukuran gambar menjadi sama.

Proses yang dilakukan pada *Image Augmentation* sebagai berikut, pertama dilakukan pembuatan variable *train_generator* dengan memanggil variable *datagen* dari direktori *train*. Kedua, ukuran target yang digunakan dalam proses training. Ketiga, mode yang dipilih adalah *categorical* dikarenakan penelitian ini menggunakan *multi class*. Keempat, banyaknya citra yang dimasukkan dalam setiap *steps training*. Kelima, data didalam folder tersebut diacak sehingga tidak sesuai urutan yang ada seperti urutan alfabetik.

```
1. train_generator = train_datagen.flow_from_directory(dir_train,
2.     target_size=(150, 150),
3.     class_mode="categorical",
4.     batch_size=32,
5.     shuffle=True
```

3. Arsitektur CNN

Pada model, API fungsional baru juga dapat dibuat dengan menggunakan *tensorflow*. Ini memungkinkan untuk mengekstrak sub-komponen model dengan cepat [9] Model yang digunakan berbentuk *sequential*, pada penelitian ini menggunakan beberapa susunan layer. Penelitian ini menggunakan kernel 3x3. Pada penelitian ini menggunakan 4 konvolusi layer dengan filter 32, 32, 64, dan 64.

Model CNN pada penelitian ini dibangun dengan library *tensorflow* untuk pembuatan model. Layer yang tersusun menggunakan filter 32 dan *kernel size* 3 x 3 dan menggunakan *stride* 1 serta, menggunakan fungsi *aktivasi relu* dan menentukan *input shape* sebesar 150 x 150 dan 3 chanel RGB. Berikutnya lakukan konvolusi dengan aktivasi *relu* juga menggunakan 32 *filter* dan terdapat *max pooling layer*, *max pooling* selalu berada setelah konvolusi layer, *max pooling layer* berfungsi untuk mereduksi resolusi gambar agar proses ketika pelatihan lebih cepat. Hasil dari *konvolusi* dilanjutkan dengan

aktivasi dengan menggunakan filter 64. Berikutnya dilakukan perataan gambar menjadi 1 dimensi yang disebut dengan *flatten*. Pada baris ke tujuh terdapat *hidden layer* dengan fungsi aktivasi yang umum digunakan untuk klasifikasi gambar yaitu fungsi aktivasi *relu* dengan variasi *perceptron* 128. Pada baris kedelapan terdapat *hidden layer* dengan *perceptron* 4 menggunakan fungsi aktivasi *softmax* untuk klasifikasi banyak kelas atau *multiclass*.

```
1. import tensorflow as tf
   model = tf.keras.models.Sequential(
2.     [
3.         tf.keras.layers.Conv2D(32, (3,3), strides=(1, 1), activation='relu', input_shape=(150, 150, 3)), #Kernel size 3 x 3
4.         tf.keras.layers.MaxPooling2D(2, 2),
5.         .....
6.         tf.keras.layers.Flatten(),
7.         tf.keras.layers.Dense(128, activation='relu'),
8.         tf.keras.layers.Dense(5, activation='softmax')
9.     ])
   model.compile(
       optimizer=tf.optimizers.Adam(learning_rate=0.002,
       loss='categorical_crossentropy',
       metrics=['accuracy'])
   model.summary()
```

4. Klasifikasi dengan CNN

Setelah dilakukan proses *preprocessing* pada data setelah terbentuknya model maka data akan di klasifikasi. Proses *training* terhadap model dilakukan dengan cara memanggil model *fit()* serta mengisi parameter diantaranya *iterasi* atau *epoch* dari pelatihan yang nantinya ditampilkan untuk melihat performa model pada saat proses training. Selanjutnya dilakukan uji model menggunakan *confusion matrix* dan *classification report* untuk mengukur performa model berupa akurasi, presisi, dan *recall*. Hasil akhir akan menampilkan akurasi pengujian data validasi.

```
1. log = model.fit(
2.     train_generator,
3.     steps_per_epoch=25,
4.     epochs=400,
5.     validation_data=validation_generator,
6.     validation_steps=6,
7.     callbacks=[cb])
```

```
Epoch 1/400
15/25 [====...] - ETA: 0s - loss: 1.1803 - accuracy: 0.4863
Epoch 1: saving model to model.h5
15/25 [====...] - ETA: 0s - loss: 1.1803 - accuracy: 0.4863 - val_loss: 1.2892 - val_accuracy: 0.5863
Epoch 2/400
15/25 [====...] - ETA: 0s - loss: 0.6987 - accuracy: 0.7262
Epoch 2: saving model to model.h5
15/25 [====...] - ETA: 0s - loss: 0.6987 - accuracy: 0.7262 - val_loss: 0.2875 - val_accuracy: 0.8562
Epoch 3/400
15/25 [====...] - ETA: 0s - loss: 0.4555 - accuracy: 0.7937
Epoch 3: saving model to model.h5
15/25 [====...] - ETA: 0s - loss: 0.4555 - accuracy: 0.7937 - val_loss: 0.3237 - val_accuracy: 0.8625
Epoch 4/400
15/25 [====...] - ETA: 0s - loss: 0.4574 - accuracy: 0.8238
Epoch 4: saving model to model.h5
15/25 [====...] - ETA: 0s - loss: 0.4574 - accuracy: 0.8238 - val_loss: 0.5085 - val_accuracy: 0.7937
Epoch 5/400
15/25 [====...] - ETA: 0s - loss: 0.5516 - accuracy: 0.7738
Epoch 5: saving model to model.h5
15/25 [====...] - ETA: 0s - loss: 0.5516 - accuracy: 0.7738 - val_loss: 0.1988 - val_accuracy: 0.9187
```

Gambar 13. Hasil Trainin

Pada Gambar 13 merupakan proses pelatihan model terhadap data *training* dan data *validation*, model menggunakan *callbacks early stopping* dengan *monitor*

Klasifikasi Jenis Jambu Biji berdasarkan tekstur daun menggunakan *Convolutional Neural Networks (CNN)*

validation loss, proses pelatihan akan terus berjalan sesuai dengan epoch yang ditentukan yakni 400 epoch.

1. *Loss* menyatakan nilai dari *loss function*, terlihat bahwa pada Gambar 8 *epoch* kesatu *loss* mendapatkan nilai yang cukup besar kemudian nilai itu menurun untuk setiap *epoch*.
2. *Accuracy* merupakan nilai akurasi dari data latih, terlihat pada Gambar 8 akurasi yang didapat 0.4836 atau bisa dibilang 48%, hal ini diharapkan terus meningkat seiring bertambahnya iterasi.
3. *Loss Validation* menyatakan nilai dari *loss function* pada data validasi, terlihat bahwa pada Gambar 8 *epoch* kesatu *loss* mendapatkan nilai yang cukup besar kemudian nilai itu *fluktuatif* untuk setiap *epoch*.
4. *Validation Accuracy val* merupakan nilai akurasi dari data validasi, terlihat pada Gambar 8 akurasi yang didapat 0.5063 atau bisa dibilang 50%, hal ini diharapkan terus meningkat seiring bertambahnya *iterasi*.

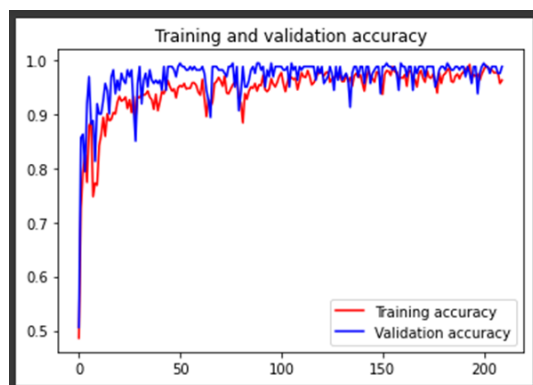
5. Pengujian Model

Pengujian model ini akan mengimplementasikan callbacks dan *earlystopping* model checkpoint dengan parameter monitor validation loss dengan menggunakan 400 epochs, ketika *loss validation* dianggap sudah optimal akan proses training akan dihentikan.

```
Epoch 200/400
25/25 [====] - ETA: 0s - loss: 0.0001 - accuracy: 0.9812
Epoch 200: saving model to model15
Epoch 200/400
25/25 [====] - ETA: 0s - loss: 0.0001 - accuracy: 0.9812 - val_loss: 0.0007 - val_accuracy: 0.9875
Epoch 200: saving model to model15
Epoch 200/400
25/25 [====] - ETA: 0s - loss: 0.0001 - accuracy: 0.9812 - val_loss: 0.0004 - val_accuracy: 0.9875
Epoch 207/400
25/25 [====] - ETA: 0s - loss: 0.0004 - accuracy: 0.9782
Epoch 207: saving model to model15
Epoch 207/400
25/25 [====] - ETA: 0s - loss: 0.0004 - accuracy: 0.9782 - val_loss: 0.0781 - val_accuracy: 0.9875
Epoch 208/400
25/25 [====] - ETA: 0s - loss: 0.0006 - accuracy: 0.9782
Epoch 208: saving model to model15
Epoch 208/400
25/25 [====] - ETA: 0s - loss: 0.0006 - accuracy: 0.9782 - val_loss: 0.0076 - val_accuracy: 0.9758
Epoch 209/400
25/25 [====] - ETA: 0s - loss: 0.0006 - accuracy: 0.9803
Epoch 209: saving model to model15
Epoch 209/400
25/25 [====] - ETA: 0s - loss: 0.0006 - accuracy: 0.9803 - val_loss: 0.0714 - val_accuracy: 0.9758
Epoch 210/400
25/25 [====] - ETA: 0s - loss: 0.0008 - accuracy: 0.9825
Epoch 210: saving model to model15
Epoch 210/400
25/25 [====] - ETA: 0s - loss: 0.0008 - accuracy: 0.9825 - val_loss: 0.0363 - val_accuracy: 0.9875
Epoch 210: early stopping
```

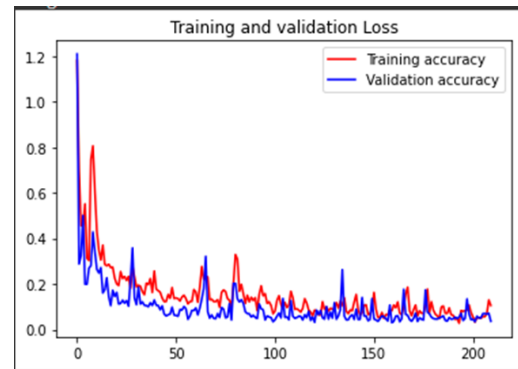
Gambar 14. Pelatihan Model

Gambar 14 merupakan hasil proses pelatihan model terhadap jumlah epoch, dapat diketahui bahwa pada proses pengujian ini menargetkan akurasi menggunakan *callbacks earlystopping* dengan monitor val los. Model berhasil mencapai nilai akurasi 96% pada epoch ke 210 dan proses pelatihan dihentikan secara otomatis.



Gambar 15. Grafik Accuracy

Pada Gambar 15 Grafik akurasi terjadi penurunan pada validation accuracy, namun akurasi kembali naik dan optimal dengan penurunan yang tidak signifikan.



Gambar 16. Grafik Lost

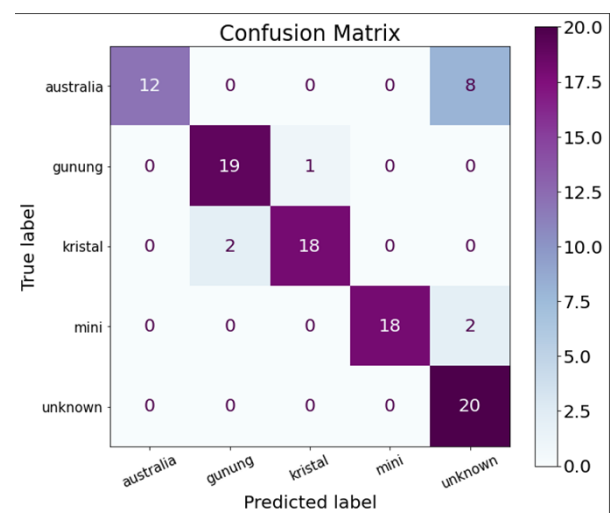
Pada Gambar 16 kurva grafik loss sudah terlihat cukup baik terdapat beberapa peningkatan yang tidak terlalu signifikan, serta mengalami penurunan ketika sampai pada iterasi ke 210.

Tabel 1 dibawah ini merupakan tabel hasil proses pelatihan menggunakan epoch 400 dapat dilihat loss training, training accuracy, loss validation, validation accuracy dan waktu komputasi per-iterasi.

Tabel 1. Hasil Akurasi

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy	Waktu Komputasi
224	0.1058	0.9625	0.0363	0.9875	6s

Gambar 11 menunjukkan hasil prediksi menggunakan Confusion matriks menggunakan data testing dengan jumlah sampel sebanyak 100 citra yang dibagi menjadi 5 class sehingga menjadi 20 citra setiap classnya.



Gambar 11. Hasil Prediksi

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP}{Total} \times 100\% \\
 &= \frac{12}{20} \times 100\% \\
 &= 60\%
 \end{aligned}$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

$$= \frac{12}{12 + 8} \times 100\%$$

$$= 60\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$= \frac{12}{12 + 0} \times 100\%$$

$$= 100\%$$

Berdasarkan perhitungan manual diatas maka akurasi untuk jenis daun jambu merah australia yang diperoleh sebesar 60%. Serta juga memperoleh nilai Recall dan Precision sebesar 60% serta 100%.

$$Akurasi = \frac{TP}{Total} \times 100\%$$

$$= \frac{19}{20} \times 100\%$$

$$= 95\%$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

$$= \frac{19}{19 + 1} \times 100\%$$

$$= 95\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$= \frac{19}{19 + 2} \times 100\%$$

$$= 90\%$$

Berdasarkan perhitungan manual diatas maka akurasi untuk jenis daun jambu gunung yang diperoleh sebesar 95%. Serta juga didapatkan recall yang serupa yakni 95%, precision yang didapatkan sebesar 90%

$$Akurasi = \frac{TP}{Total} \times 100\%$$

$$= \frac{18}{20} \times 100\%$$

$$= 90\%$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

$$= \frac{18}{20} \times 100\%$$

$$= 90\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$= \frac{18}{18 + 1} \times 100\%$$

$$= 95\%$$

Berdasarkan perhitungan manual diatas maka akurasi dan recall untuk jenis daun jambu kristal yang diperoleh sebesar 90%. Serta diperoleh precision sebesar 95%.

$$Akurasi = \frac{TP}{Total} \times 100\%$$

$$= \frac{18}{20} \times 100\%$$

$$= 90\%$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

$$= \frac{18}{18 + 2} \times 100\%$$

$$= 95\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$= \frac{18}{18 + 0} \times 100\%$$

$$= 100\%$$

Berdasarkan perhitungan manual diatas maka akurasi untuk jenis daun jambu mini yang diperoleh sebesar 90% serta menghasilkan nilai recall dan precision yang serupa yakni 95% dan 100%.

Classification Report				
	precision	recall	f1-score	support
0	1.00	0.60	0.75	20
1	0.90	0.95	0.93	20
2	0.95	0.90	0.92	20
3	1.00	0.90	0.95	20
4	0.67	1.00	0.80	20
accuracy			0.87	100
macro avg	0.90	0.87	0.87	100
weighted avg	0.90	0.87	0.87	100

Gambar 11. Classification Report

Untuk menghitung total akurasi yang diperoleh dari data uji atau data test dapat menggunakan rumus accuracy sehingga diperoleh hasil sebagai berikut.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

$$= \frac{12 + 19 + 18 + 18 + 20}{100} \times 100\%$$

$$= 87\%$$

Berdasarkan perhitungan diatas maka hasil akurasi yang diperoleh menggunakan data testing yaitu sebesar 87%.

Klasifikasi Jenis Jambu Biji berdasarkan tekstur daun menggunakan *Convolutional Neural Networks (CNN)*

Tabel 2 dibawah ini merupakan hasil dari pengujian terhadap training dengan 400 epoch. Digunakan 20 gambar data uji dari data test pada masing-masing kelas untuk dilakukan pengujian. Lalu untuk menentukan akurasi dari data testing maka dilakukan perhitungan menggunakan rumus akurasi.

Tabel 2 Hasil Akurasi Menggunakan Data Testing

Label	Jumlah Data Uji	Prediksi Benar	Prediksi Salah	Akurasi
Daun Jambu Merah Australia	20	12	8	60%
Daun Jambu Gunung	20	19	1	95%
Daun Jambu Kristal	20	18	2	90%
Daun Jambu Mini	20	18	2	90%
Unknown	20	20	0	100%

6. Interpretasi Hasil

Setelah dilakukan Pengujian model mengimplementasikan *epoch* yang besar dan menggunakan *callbacks*. Hal ini bertujuan untuk mengetahui kinerja model dan mencari model yang ideal. Hasil terbaik pada pengujian menggunakan data testing mendapat nilai akurasi sebesar 95% yaitu daun jambu gunung. Sedangkan hasil terkecil mendapat nilai akurasi sebesar 60% yaitu daun jambu merah australia.

V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan bahwa “Klasifikasi Jenis Jambu Berdasarkan Warna Dan Tekstur Daun Menggunakan Metode Convolutional Neural Networks” berhasil mengklasifikasi jenis jambu berdasarkan daun dengan baik atau dapat mengenali jenis jambu berdasarkan daun menggunakan data yang baru. Hasil akurasi yang didapat menggunakan data training dengan jumlah epochs 400 menghasilkan akurasi sebesar 96% pada iterasi ke 210. Sedangkan hasil yang diperoleh menggunakan data test atau data baru yang tidak terdapat pada data training dengan menggunakan 100 gambar yang masing masing terbagi menjadi empat kelas sehingga 20 gambar pada tiap kelas memperoleh hasil akurasi sebesar 87%. Sehingga pengklasifikasian pada penelitian ini sudah memuaskan hasil yang baik, model sudah dapat mengenali objek dengan baik menggunakan data baru.

VI. DAFTAR PUSTAKA

- [1] E. Díaz-de-Cerio, V. Verardo, A. M. Gómez-Caravaca, A. Fernández-Gutiérrez, and A. Segura-Carretero, *Health effects of Psidium guajava L. Leaves: An overview of the last decade*, vol. 18, no. 4, 2017. doi: 10.3390/ijms18040897.
- [2] S. N. Khairunnisa, “15 Jenis Jambu Biji Populer di Indonesia, Dari Jambu Bangkok sampai Pasar Minggu,” *www.kompas.com/*, 2021.
- [3] A. K. Hasibuan, “Klasifikasi jenis buah jambu berdasarkan daun menggunakan metode principal component analysis,” 2020.
- [4] M. Ulil and A. Kamal, “Calculation of Handwriting Mathematics Expressions on Mobile Devices Using Efficientdet- Lite0 And Reverse Polish Notation,” pp. 1–7.
- [5] F. Handayani, R. Sundu, and R. M. Sari, “FORMULASI DAN UJI AKTIVITAS ANTIBAKTERI Streptococcus mutans DARI SEDIAAN MOUTHWASH EKSTRAK DAUN JAMBU BIJI (Psidium guajava L .) PENDAHULUAN Penyakit karies gigi dan jaringan pendukung gigi (periodontal) umumnya disebabkan oleh plak gigi , yang sampai,” vol. 1, no. 8, pp. 422–433, 2017.
- [6] P. H. N. Rusdi, F. Oenzil, and E. Chundrayetti, “Pengaruh Pemberian Jus Jambu Biji Merah (Psidium Guajava.L) Terhadap Kadar Hemoglobin dan Ferritin Serum Penderita Anemia Remaja Putri,” *J. Kesehat. Andalas*, vol. 7, no. 1, p. 74, 2018, doi: 10.25077/jka.v7.i1.p74-79.2018.
- [7] Fadhillah, Annisa, Susanti, Sri, Gultom, and Tumiur, “Karakterisasi Tanaman Jambu Biji (Psidium guajava L.) di Desa Namoriam Pancur Batu Kabupaten Deli Serdang Sumatera Utara,” *Pros. Semin. Nas. Biol. dan Pembelajarannya*, p. 1670, 2018.
- [8] M. R. Prasanta, M. Y. Pranata, M. A. Firnanda, and S. Sendari, “Rancang Bangun Quadcopter Drone Untuk Deteksi Api Menggunakan YOLOv4,” *Cyclotron*, vol. 5, no. 1, 2022, doi: 10.30651/cl.v5i1.10013.
- [9] <https://www.tensorflow.org/>, “TensorFlow,” 2022. <https://www.tensorflow.org/resources/tools>
- [10] K. Daun *et al.*, “Tea Leaves Gmb Series Clasiffication Using Convolutional Neural Network,” vol. 3, no. 2, pp. 0–5, 2020.
- [11] K. Kartarina, L. Z. A. Mardedi, M. Madani, M. Jihad, and R. A. Riberu, “Deep Learning Identifikasi Tanaman Obat Menggunakan Konsep Siamese Neural Network,” *JTIM J. Teknol. Inf. dan Multimed.*, vol. 2, no. 4, pp. 223–228, 2021, doi: 10.35746/jtim.v2i4.114.
- [12] Wayan Dadang, “Memahami Kecerdasan Buatan berupa Deep Learning dan Machine Learning,” <https://warstek.com/>, 2021. <https://warstek.com/deepmachinelearning/mathworks>
- [13] “Convolutional Neural Network,” <https://www.mathworks.com/>, 2020. <https://www.mathworks.com/discovery/convolutional-neural-network-matlab.html>
- [14] S. Ilahiyah and A. Nilogiri, “Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network,” *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.)*, vol. 3, no. 2, pp. 49–56, 2018.
- [15] I. N. Laily, “Pengertian Penelitian Kuantitatif, Karakteristik dan Jenisnya,” <https://katadata.co.id/>,

2022.

<https://katadata.co.id/iftitah/ekonopedia/6295749c7fdd7/pengertian-penelitian-kuantitatif-karakteristik-dan-jenisnya#:~:text=Cresswell>