

# KLASIFIKASI JENIS PENYAKIT PADA DAUN TUMBUHAN STROBERI MENGGUNAKAN METODE *CONVOLUTIONAL NEURAL NETWORK* ARSITEKTUR *INCEPTIONV3*

Andrian Herbert Parsaoran Sitohang<sup>1\*</sup>, Teguh Iman Hermanto<sup>2</sup>, Candra Dewi Lestari<sup>3</sup>

<sup>1</sup>Sekolah Tinggi Teknologi Wastukencana Purwakarta, Indonesia

<sup>2</sup>Sekolah Tinggi Teknologi Wastukencana Purwakarta, Indonesia

<sup>3</sup>Sekolah Tinggi Teknologi Wastukencana Purwakarta, Indonesia

Received: 28 Agustus 2024

Accepted: 5 Oktober 2024

Published: 12 Oktober 2024

## Keywords:

*Convolutional Neural Network*; Daun stroberi; *InceptionV3*; Klasifikasi Penyakit.

## Correspondent Email:

[andrianherbert50@wastuka.ncana.ac.id](mailto:andrianherbert50@wastuka.ncana.ac.id)

**Abstrak.** Stroberi adalah salah satu komoditas tanaman dengan nilai ekonomi yang cukup tinggi di Indonesia. Namun, terdapat upaya yang dapat mengurangi kualitas dan kuantitas hasil panen stroberi, yaitu penyakit-penyakit daun stroberi, termasuk *leaf scorch*. Oleh karena itu, deteksi dini dan klasifikasi penyakit mengenai daun stroberi ini sangat penting untuk melakukan tindakan yang diperlukan agar kerugian dapat diminimalkan. Tujuan dari penelitian ini adalah untuk mengembangkan model klasifikasi penyakit pada daun stroberi dengan metode *Convolutional Neural Network* menggunakan arsitektur *InceptionV3*. Proses penelitian yang dibutuhkan dalam pengembangan model melibatkan pengumpulan *dataset* gambar daun stroberi yang terdiri dari kategori sehat dan penyakit *leaf scorch*. Model CNN dilatih dan diuji dengan menggunakan *dataset* gambar daun tersebut untuk mengidentifikasi dan mengklasifikasikan jenis penyakit dengan tingkat akurasi yang tinggi. Hasil dari penelitian ini menunjukkan bahwa model CNN dengan arsitektur *InceptionV3* berhasil mencapai akurasi klasifikasi sebesar 99%, menandakan keandalan model dalam mendeteksi penyakit pada daun stroberi. Selain itu, Implementasi model ini dalam aplikasi berbasis Android diharapkan dapat memberikan alat bantu yang praktis dan efisien bagi petani dalam memantau dan mengendalikan penyakit tanaman secara real-time, sehingga dapat meningkatkan kualitas dan kuantitas hasil panen stroberi. Dengan demikian, penelitian ini tidak hanya berkontribusi pada pengembangan teknologi deteksi penyakit tanaman tetapi juga pada peningkatan produktivitas pertanian.

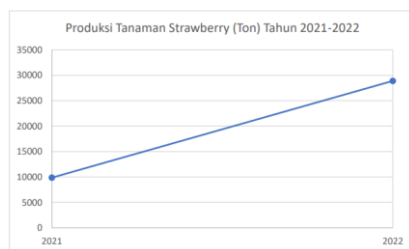
**Abstract.** Strawberries are one of the crop commodities with quite high economic value in Indonesia. However, there are efforts that can reduce the quality and quantity of strawberry harvests, namely strawberry leaf diseases, including leaf scorch. Therefore, early detection and classification of these diseases regarding strawberry leaves is very important to take the necessary measures so that losses can be minimized. The purpose of this study is to develop a disease classification model in strawberry leaves using the Convolutional Neural Network method using the InceptionV3 architecture. The research process needed in the development of the model involves the collection of a dataset of strawberry leaf images consisting of healthy categories and leaf scorch diseases. The CNN model was trained and tested using a dataset of leaf images to identify and classify disease types with a high degree of accuracy. The results of this study show that the CNN model with the InceptionV3 architecture has managed to achieve a classification

*accuracy of 99%, indicating the reliability of the model in detecting diseases in strawberry leaves. In addition, the implementation of this model in an Android-based application is expected to provide practical and efficient tools for farmers in monitoring and controlling plant diseases in real-time, so as to improve the quality and quantity of strawberry crops. Thus, this research not only contributes to the development of plant disease detection technology but also to the increase of agricultural productivity.*

## 1. PENDAHULUAN

Stroberi merupakan tanaman buah dengan rasanya yang cukup manis, warna dan bentuk buah yang menarik, serta komposisi gizi, vitamin, dan anti oksidan yang tinggi [1]. Stroberi adalah tanaman subtropis yang apabila ditanam di daerah tropis pertumbuhannya akan optimal bila ditanam pada dataran tinggi dengan suhu 14-24 derajat *celcius* [2]. Maka dari itu, budidaya stroberi di Indonesia banyak diusahakan di dataran tinggi seperti Ciwidey, Lembang, Sukabumi, dan Cianjur [1]. Selain itu, tanaman stroberi merupakan komoditas pertanian yang bernilai ekonomis tinggi dan memiliki prospek usaha yang menjanjikan [3].

Menurut data yang dikeluarkan oleh Badan Pusat Statistik, jumlah produksi tanaman stroberi di Indonesia mengalami peningkatan yang sangat signifikan, tercatat pada tahun 2021 jumlah produksi stroberi mencapai angka 9860 Ton, sedangkan pada tahun 2022 jumlah produksi stroberi mencapai angka 28895 Ton [4].



Gambar 1. Grafik Produksi Tanaman Stroberi  
Sumber : BPS Indonesia, 2023

Namun, dalam proses perkembangannya, masih terjadi kendala karena teknik budidaya yang digunakan masih bersifat konvensional, sehingga hasil yang diperoleh belum mampu memenuhi permintaan pasar yang ada. Permasalahan utama yang dihadapi dalam budidaya stroberi di Indonesia adalah keterbatasan dalam penyediaan bibit yang berkualitas tinggi dan bebas dari penyakit [3]. Para petani yang masih menggunakan Teknik

budidaya konvensional akan kesulitan untuk mengidentifikasi penyakit-penyakit yang menjangkit tanaman stroberinya, hal itu karena untuk dapat mengidentifikasi penyakit pada tanaman stroberi biasanya dilakukan oleh para pakar tumbuhan.

Sehingga untuk menjaga tren kenaikan produksi tanaman stroberi disetiap tahunnya, maka para petani di Indonesia perlu dapat mengidentifikasi penyakit-penyakit yang menjangkit tanaman stroberi dengan lebih cepat, mudah, dan akurat. Salah satu inovasi yang mungkin dapat diterapkan dalam upaya mengidentifikasi atau mengklasifikasikan kondisi kesehatan tanaman stroberi dengan lebih mudah dan akurat adalah melalui penggunaan teknologi pengenalan citra daun stroberi dengan metode *Convolutional Neural Network* (CNN). CNN sendiri merupakan salah satu model pembelajaran *deep learning* yang mampu melatih sistem dengan data yang cukup banyak serta menggabungkan proses ekstraksi ciri dan klasifikasi [5].

Sebelumnya sudah banyak penelitian yang menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur *InceptionV3*, seperti penelitian klasifikasi warna pada kematangan buah kopi kuning didapatkan nilai akurasi metode CNN *InceptionV3* nya sebesar 92.00% dengan total 1380 *dataset* citra buah kopi yang dipecah menjadi 984 citra 12 untuk *training*, 246 citra untuk validasi dan 150 citra buah kopi untuk *testing* [6].

Penelitian "Implementasi *Convolutional Neural Networks* dengan *InceptionV3* untuk Deteksi Katarak Menggunakan Citra Funduskopi Digital" yang dilakukan oleh M. Ahnaf Amrullah dan M. Isa Irawan. *Contrast Limited Adaptive Histogram Equalization* (CLAHE) pada kanal hijau dan *histogram equalization* digunakan sebagai teknik pra-pemrosesan data pencitraan fundus. Model dengan *preprocessing* CLAHE dan *fine tuning*,

yang memiliki akurasi 98,33%, memberikan hasil terbaik dalam investigasi ini [7].

Peneliti Muhammad Baihaqy, Agung Toto Wibowo, dan Dody Qori Utama melakukan penelitian yang berjudul "Klasifikasi Tanaman Anggrek *Phalaenopsis* Berdasarkan Citra Labellum Bunga Menggunakan Metode *Convolutional Neural Network* (CNN)". Penelitian ini mengembangkan sebuah sistem berbasis *Convolutional Neural Network* (CNN) yang efektif mengklasifikasikan gambar labellum pada bunga anggrek. Dengan menggunakan arsitektur *InceptionV3* dan skema tiga kali uji coba, penelitian ini menemukan bahwa uji coba ketiga, yang menggunakan fold kelima, memiliki nilai akurasi tertinggi, dengan rata-rata akurasi 100%. Desain ini memiliki recall rata-rata 98,6%, nilai presisi rata-rata 99,89%, nilai f1 rata-rata 99,2%, dan nilai akurasi rata-rata 99,2% di semua pengujian [8].

Selanjutnya penelitian dengan judul "Penerapan Metode *Convolutional Neural Network* (CNN) Dalam Mengklasifikasikan Penyakit Daun Tanaman Padi" yang dilakukan oleh Gracia, dkk. Penelitian ini melakukan perbandingan jumlah *epoch* serta menggunakan arsitektur CNN *InceptionV3*. Hasil dari penelitian ini menunjukkan hasil yang sangat baik di angkat 98% dengan data yang tidak overfitting [9].

Penelitian ini bertujuan untuk membangun dan mengimplementasikan model klasifikasi menggunakan *Convolutional Neural Network* (CNN) dengan arsitektur *InceptionV3* untuk mendeteksi dan mengidentifikasi berbagai penyakit pada daun tanaman stroberi. Selain itu, model ini dirancang agar dapat digunakan secara langsung oleh petani melalui aplikasi *mobile* berbasis Android.

## 2. TINJAUAN PUSTAKA

### 2.1. Klasifikasi Citra

*Image classification* adalah klasifikasi gambar dengan cara mengelompokkan objek dan menkonversinya menjadi angka yang ditentukan sehingga dapat mudah mengenali objek. Proses ini diawali dengan *image processing* untuk memperjelas data gambar yang akan dibaca oleh mesin [10].

### 2.2. *Convolutional Neural Network*

Arsitektur dari metode *Convolutional Neural Network* ada banyak variasi salah

satunya adalah *InceptionV3*. Berawal dari *GoogleNet* yang diberi nama *InceptionV1*. Kemudian disempurnakan lagi melalui berbagai cara dengan pengenalan *batch normalization* yang dapat disebut *InceptionV2*. Lanjut pengembangan dari *InceptionV2* menjadi *InceptionV3* dengan penambahan ide faktorisasi di iterasi ketiga. Model arsitektur ini menggunakan dimensi yang mereduksi pada setiap tahapan proses, sehingga waktu komputasi yang dilakukan lebih cepat dan alokasi memori yang digunakan baik dan dapat memberikan performa yang tinggi dalam klasifikasi citra gambar meskipun dengan jumlah memori yang terbatas [11].

### 2.3. CRISP-DM

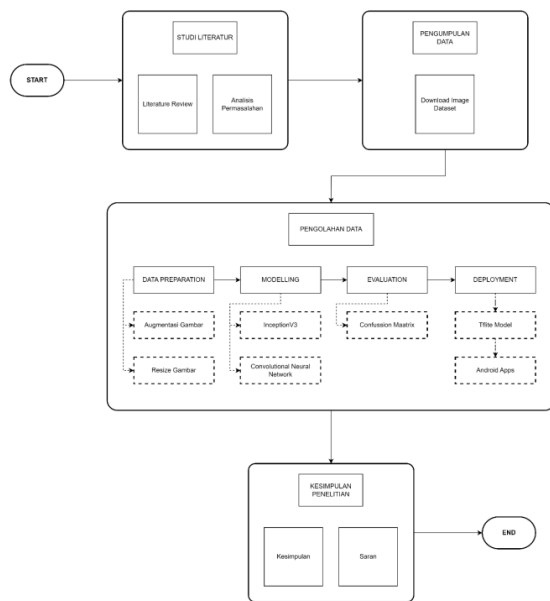
*Model Cross-Industry Standard Process for Data mining* (CRISP-DM) adalah metode yang digunakan untuk mengelola proses pengolahan data (data mining) dari awal hingga akhir [12]. Model ini dikembangkan oleh Uni Eropa dan melibatkan enam tahap utama: Pemahaman Bisnis (*Business Understanding*), Pemahaman Data (*Data understanding*), Persiapan Data (*Preparation*), Pemodelan (*Modeling*), Evaluasi (*Evaluation*), dan Penerapan (*Deployment*) [13] [14].



Gambar 2. Tahapan CRISP-DM  
Sumber : Yudiana

## 3. METODE PENELITIAN

Metodologi penelitian yang akan dilakukan untuk klasifikasi jenis penyakit pada daun tumbuhan stroberi menggunakan metode *Convolutional Neural Network* arsitektur *InceptionV3* sebagai berikut:



Gambar 3. Metodologi Penelitian

### 3.1. Study Literatur

Berdasarkan latar belakang masalah yang telah dirumuskan berdasarkan data pendukung dan didukung oleh jurnal dan penelitian terdahulu, maka dapat disimpulkan bahwa tujuan penelitian ini dilakukan untuk membuat model yang dapat mengenali citra daun tanaman stroberi untuk diklasifikasikan berdasarkan kondisi kesehatannya. Klasifikasi citra menggunakan metode *Convolutional Neural Network* dengan arsitektur *InceptionV3*.

### 3.2. Pengumpulan Data

Data yang dibutuhkan untuk membuat model yang dapat mengenali citra daun tanaman stroberi untuk diklasifikasikan berdasarkan kondisi kesehatannya yaitu berupa citra daun tanaman stroberi. Data tersebut penulis dapatkan dalam bentuk *dataset* berisi kumpulan foto yang berasal dari situs kaggle.com dengan ukuran keseluruhan dari *dataset* ini adalah sebesar 798MB dengan jumlah total data citra sebanyak 53.300 data citra. Masing-masing citra pada *dataset* memiliki 3 channel yaitu *red*, *green*, dan *blue*. *Sampel* dari data citra pada *dataset* dapat dilihat pada gambar 4 dibawah.



Gambar 4. Sample Data Citra

### 3.3. Data Preparation

Proses data *Preparation* adalah proses dimana penulis akan mengolah *dataset* yang telah didapatkan. Proses pengolahan bertujuan untuk memudahkan model memproses gambar juga untuk meningkatkan kinerja model.

#### 3.3.1. Augmentasi

Augmentasi adalah proses dimana masing-masing gambar akan diperbanyak dengan cara merubah sudut pandang gambar (*flip*), kemiringan gambar (*rotation*), dan memperbesar focus gambar (*zoom*). Proses ini bertujuan agar model dapat mengenali gambar dalam berbagai posisi.

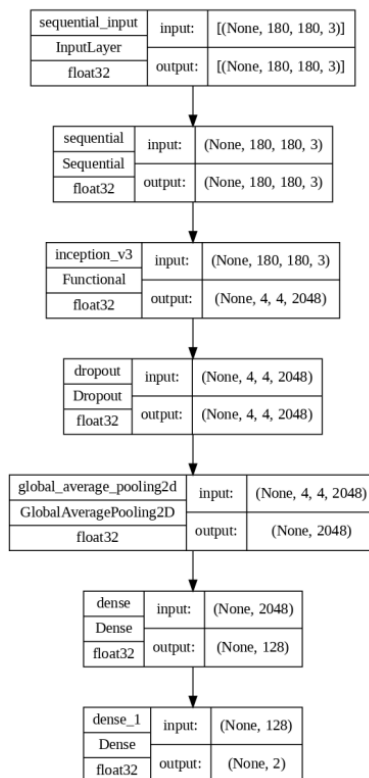
#### 3.3.2. Resize Data

*Dataset* yang telah didapatkan berisi kumpulan citra daun tanaman stroberi dengan berbagai ukuran. Keberagaman ukuran ini akan membuat hasil klasifikasi model menjadi tidak konsisten karena model akan mengkalkulasi nilai *pixel* dalam citra dengan jumlah yang berbeda-beda. Maka dari itu seluruh citra pada *dataset* perlu untuk disamaratakan ukurannya menjadi 180x180.

### 3.4. Modelling

Dalam langkah ini, penulis melakukan pembuatan sebuah model yang akan dipakai untuk mengklasifikasikan gambar daun dari tanaman stroberi. Model yang dirancang akan menggunakan metode *Convolutional Neural Networks* dengan mengadopsi arsitektur *InceptionV3*. Rancangan alur model tersebut dapat ditemukan dalam gambar di bawah ini.





Gambar 5. Rancangan Alur Model

### 3.5. Evaluation

Tahap ini bertujuan untuk menilai kinerja model yang telah berhasil dibangun. Evaluasi dilakukan dengan menggunakan Metode Matriks Confusion untuk melihat bagaimana model berperilaku terhadap seluruh dataset. Selain itu, eksperimen dilakukan dengan menggunakan gambar yang telah disiapkan untuk menguji apakah model mampu melakukan klasifikasi dengan akurat atau tidak.

### 3.6. Deployment

Tahap deployment merupakan tahapan di mana model yang telah dikembangkan melalui tahap evaluasi berhasil mencapai kinerja optimal. Selanjutnya, model tersebut akan diubah formatnya menjadi *tensorflow lite* agar dapat diaplikasikan pada berbagai *platform*, baik *desktop* maupun *mobile*. Langkah selanjutnya adalah menerapkan model tersebut ke dalam aplikasi *mobile* berbasis Android untuk keperluan implementasi.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Pengumpulan Data

Setelah berhasil mengunduh *dataset* dari situs Kaggle.com, *dataset* selanjutnya akan dibagi menjadi dua kategori yaitu data *training*

dan data *testing*. Dalam pembagian *dataset* menggunakan rasio 8:2 atau 80% untuk data *training* dan 20% untuk data *testing*.

```
[ ] train_ds = tf.keras.utils.image_dataset_from_directory(
    data_dir,
    validation_split=0.2,
    subset="training",
    seed=123,
    image_size=(img_height, img_width),
    batch_size=batch_size)
```

Found 6146 files belonging to 2 classes.  
Using 4917 files for training.

```
[ ] val_ds = tf.keras.utils.image_dataset_from_directory(
    data_dir,
    validation_split=0.2,
    subset="validation",
    seed=123,
    image_size=(img_height, img_width),
    batch_size=batch_size)
```

Found 6146 files belonging to 2 classes.  
Using 1229 files for validation.

Gambar 6. Source Code Pembagian Data Training dan Data Testing

Dengan menggunakan baris kode diatas, penulis membagi *dataset* menjadi dua kategori yaitu data *training* sebanyak 4917 data atau 80% dari total keseluruhan *dataset* dan data *testing* sebanyak 1229 data atau 20% dari total keseluruhan dataset. Pada proses ini juga, terdapat parameter yang berfungsi mengatur Batch\_size untuk model. Batch\_size adalah jumlah gambar dalam satu kelompok, dimana model nantinya akan mempelajari *dataset* berdasarkan kelompok.

### 4.2. Data Preparation

Setelah proses pembagian *dataset* menjadi dua kategori selesai dilakukan, tahapan selanjutnya adalah proses penyiapan dataset. Proses penyiapan *dataset* ini 32 diperlukan guna mempermudah model untuk mempelajari dataset. Pada tahapan ini, terdapat 2 kegiatan yaitu Augmentasi data dan *Resize* data.

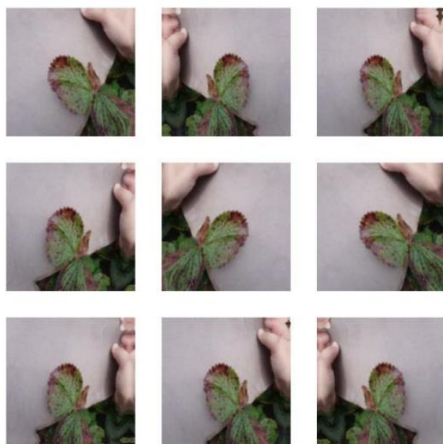
#### 4.2.1. Augmentasi

Augmentasi adalah kegiatan dimana seluruh *dataset* akan disalin untuk diperbanyak dengan berbagai sudut pandang. Dalam proses Augmentasi, seluruh data gambar diduplikasi dengan cara dirotasi dengan berbagai *angle* juga akan di zoom untuk memperjelas detail gambar.

```
[ ] data_augmentation = keras.Sequential(
    [
        layers.RandomFlip("horizontal",
                           input_shape=(img_height,
                                         img_width,
                                         3)),
        layers.RandomRotation(0.1),
        layers.RandomZoom(0.1),
    ]
)
```

Gambar 7. Source Code Untuk Augmentasi

Dengan menggunakan baris kode pada gambar 7 diatas, *dataset* akan diduplikasi menggunakan parameter random rotation dengan nilai 0.1 dan random zoom dengan nilai 0.1.



Gambar 8. Hasil Proses Augmentasi

#### 4.2.2. Resize Data

Kegiatan *Resize* data adalah kegiatan dimana seluruh gambar pada *dataset* akan dirubah ukurannya menjadi seragam yaitu 180x180. Kegiatan ini bertujuan agar model dapat bekerja dengan konsisten dan menghindari kerancuan proses klasifikasi karena perbedaan ukuran piksel gambar. Kode baris untuk kegiatan *Resize* data dapat dilihat pada gambar 6 diatas. Pada kode tersebut terdapat parameter *image\_size* yang diisi ukuran yang akan diterapkan pada *dataset*. Pada gambar 9 dibawah adalah kode baris untuk memverifikasi ukuran gambar pada *dataset*.

```
[ ] for image_batch, labels_batch in train_ds:
    print(image_batch.shape)
    print(labels_batch.shape)
    break

(32, 180, 180, 3)
(32,)
```

Gambar 9. Hasil *Resize* Data

### 4.3. Modelling

Setelah proses penyiapan data selesai dilakukan, selanjutnya adalah tahapan pembangunan model yang dapat mengklasifikasikan jenis penyakit pada tumbuhan stroberi. Proses Pembangunan model ini menggunakan metode *Convolutional Neural Network* dengan arsitektur *InceptionV3* yang akan di-import dari *library* Keras.

```
[ ] base_model = keras.applications.InceptionV3(
    include_top=False,
    weights='imagenet',
    input_shape=(180,180,3),
)

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-applications/inception_v3_weights_tf_dim_ordering_tf_kernels_notop.h5
67949568/67949568 [=====] - 35 0s/170p
```

Gambar 10. Source Code Untuk Import Arsitektur

Arsitektur *InceptionV3* sendiri memiliki ukuran sebesar 83,17 *Megabyte* dimana didalamnya terdapat total parameter sebanyak 21802784 parameter dengan 21768352 parameter sebagai trainable parameter dan 34432 sebagai non-trainable parameter.

Setelah berhasil mendapatkan arsitektur *InceptionV3*, selanjutnya arsitektur tersebut akan diimplementasikan kedalam suatu model jaringan syaraf tiruan. Pada pembuatan model jaringan syaraf tiruan menggunakan beberapa layer yang bekerja secara berurutan. Pada layer pertama adalah layer input dimana pada layer ini model jaringan syaraf tiruan akan menerima data gambar dengan ukuran 180x180, selanjutnya adalah layer augmentasi dimana data gambar yang masuk akan di augmentasi untuk diperbanyak dan dirubah sudut pandang, setelah itu data gambar akan masuk kedalam arsitektur *InceptionV3* yang menghasilkan data gambar dengan ukuran 4x4 dengan jumlah *channel* 2048, selanjutnya data gambar akan memasuki *layer dropout* yang akan membuang informasi data yang tidak relevan sehingga menghindari overfitting pada model, setelah itu data gambar akan memasuki layer *Global Average Pooling 2d* yang akan mengambil nilai rata-rata *pixel* pada data gambar, dan pada layer terakhir adalah layer dense dimana data gambar akan diklasifikasikan kedalam 2 kategori yaitu 'Strawberry Healthy' dan 'Strawberry Leaf scorch'.

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
sequential (Sequential)	(None, 180, 180, 3)	0
inception_v3 (Functional)	(None, 4, 4, 2048)	21802784
dropout (Dropout)	(None, 4, 4, 2048)	0
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dense (Dense)	(None, 128)	262272
dense_1 (Dense)	(None, 2)	258

=====  
 Total params: 22065314 (84.17 MB)  
 Trainable params: 22030882 (84.04 MB)  
 Non-trainable params: 34432 (134.50 KB)

Gambar 11. Model Jaringan Syaraf Yang Telah Berhasil Dibangun

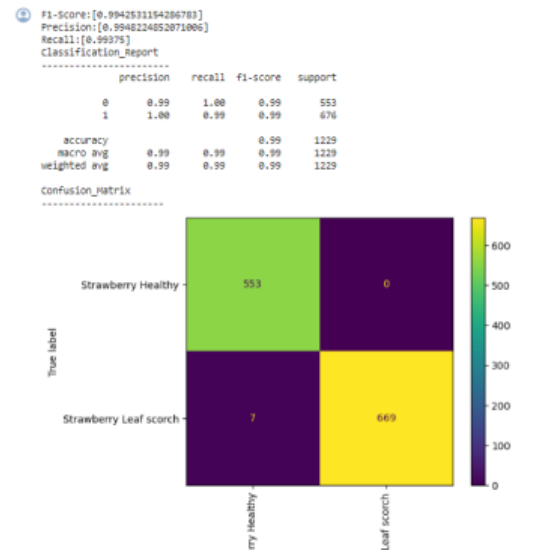
Setelah model dibangun, selanjutnya model akan dilatih untuk mengenali *dataset* baik data *training* maupun data *testing*. Proses pengenalan *dataset* ini menggunakan suatu metode bernama *epoch*. *Epoch* adalah metode perhitungan dimana ketika model bekerja pada seluruh *dataset*, disini penulis akan membuat 10 *epoch* yang dapat diartikan bahwa model akan bekerja pada seluruh *dataset* dalam 10 kali pengulangan. Pengulangan diperlukan agar model dapat mengenali data gambar pada *dataset* dengan lebih baik.

```
Epoch 1/10 [=====] - loss: 0.8639 - accuracy: 0.9819 - val_loss: 128.0858 - val_accuracy: 0.4988
Epoch 2/10 [=====] - loss: 0.8278 - accuracy: 0.9939 - val_loss: 3.2296 - val_accuracy: 0.5712
Epoch 3/10 [=====] - loss: 0.8243 - accuracy: 0.9937 - val_loss: 3.3854 - val_accuracy: 0.8723
Epoch 4/10 [=====] - loss: 0.8188 - accuracy: 0.9969 - val_loss: 0.1287 - val_accuracy: 0.9496
Epoch 5/10 [=====] - loss: 0.8002 - accuracy: 0.9978 - val_loss: 0.5486 - val_accuracy: 0.9239
Epoch 6/10 [=====] - loss: 0.8097 - accuracy: 0.9980 - val_loss: 21.6069 - val_accuracy: 0.5151
Epoch 7/10 [=====] - loss: 0.8123 - accuracy: 0.9978 - val_loss: 0.8178 - val_accuracy: 0.9943
Epoch 8/10 [=====] - loss: 0.8063 - accuracy: 0.9980 - val_loss: 18.6978 - val_accuracy: 0.4630
Epoch 9/10 [=====] - loss: 0.8059 - accuracy: 0.9980 - val_loss: 0.8118 - val_accuracy: 0.9959
Epoch 10/10 [=====] - loss: 0.8025 - accuracy: 0.9994 - val_loss: 0.8246 - val_accuracy: 0.9943
```

Gambar 12. Proses Pelatihan Model

#### 4.4. Evaluation

Setelah model jaringan syaraf tiruan berhasil dibangun, tahapan selanjutnya adalah menguji kinerja model tersebut. Pengujian model akan dilakukan dengan cara mengklasifikasi data *testing* dan akan dievaluasi menggunakan metode *confussion matrix*. Pada tahapan ini pula, model akan dinilai tingkat akurasinya.



Gambar 13. Hasil evaluasi model dengan *confussion matrix*

Pada gambar 13 diatas, terlihat bahwa model telah bekerja dengan baik dimana model mendapatkan skor akurasi sebesar 0,99%. Selain itu jika melihat pada *confussion matrix*, terlihat model lebih dominan mengklasifikasikan dengan tepat 36 dibandingkan salah sehingga dapat disimpulkan bahwa model yang telah dibangun dapat bekerja dengan baik dan optimal.

#### 4.5. Deployment

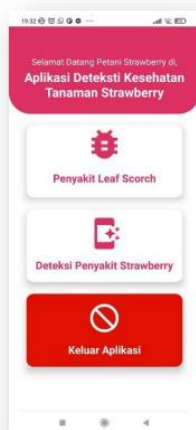
Model yang telah dinilai baik dan optimal ditahap *Evaluation* akan disimpan dalam format *tflite*. Model yang disimpan dalam format *tf-lite* dapat diimplementasikan kedalam berbagai media, salah satunya adalah media aplikasi berbasis *mobile* khususnya Android.

```
[ ] # Convert the model
converter = tf.lite.TFLiteConverter.from_keras_model(model) # path to the SavedModel directory
tflite_model = converter.convert()

# Save the model.
with open('model_ucok.tflite', 'wb') as f:
    f.write(tflite_model)
```

Gambar 14. Baris kode simpan model kedalam *tflite*

Setelah model berhasil disimpan dalam format *tflite*, selanjutnya model akan diimplementasikan kedalam suatu aplikasi Android. Dengan begitu, model dapat digunakan dengan lebih mudah dan ramah pengguna.



Gambar 15. Halaman awal aplikasi

Gambar 15 merupakan halaman awal pada aplikasi pendeteksian penyakit tanaman stroberi. Pada tampilan awal ini ada 3 pilihan tombol yaitu : penyakit *leaf scorch*, deteksi penyakit stroberi, dan keluar aplikasi.



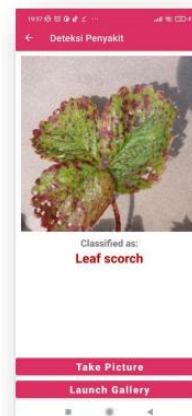
Gambar 16. Halaman Penjelasan

Jika kita mengklik tombol penyakit *leaf scorch* pada halaman awal maka akan menampilkan sebuah tampilan seperti pada gambar 16 dimana pada halaman ini akan menjelaskan lebih detail mengenai penyakit *leaf scorch* yang menyerang tanaman stroberi.



Gambar 17. Halaman Deteksi Sehat

Gambar 17 adalah halaman untuk mendeteksi penyakit pada tanaman stroberi. Di dalam halaman ini petani bisa menangkap gambar secara langsung atau meng-upload nya dari file gallery yang dari perangkat android yang mereka punya.



Gambar 18. Halaman Deteksi

Dan halaman 18 adalah halaman hasil dari pendeteksian pada tanaman stroberi yang sebelumnya telah dilakukan halaman ini akan menampilkan apakah tanaman itu sehat atau justru terkena penyakit *leaf scorch*.

## 5. KESIMPULAN

Penelitian ini menunjukkan bahwa metode *Convolutional Neural Network (CNN)* dengan arsitektur *InceptionV3* sangat efektif untuk klasifikasi penyakit pada daun tanaman stroberi, khususnya penyakit *leaf scorch*. Model yang dikembangkan berhasil mencapai akurasi tinggi sebesar 99%, yang menandakan



keandalan model dalam mendeteksi dan mengklasifikasikan penyakit. Implementasi model ini ke dalam aplikasi berbasis Android diharapkan dapat memberikan alat yang praktis bagi petani untuk pemantauan dan pengendalian penyakit tanaman, sehingga dapat meningkatkan kualitas dan kuantitas hasil panen stroberi.

#### DAFTAR PUSTAKA

- [1] M. R. Setiawati, N. Rachelita, B. N. Fitriatin, and A. Nurbaity, "Pengaruh Pemberian Asam Humat, Asam Fulvat, dan Pupuk Hayati pada Media Tanam terhadap Beberapa Sifat Kimia Tanah, Hasil, dan Kualitas Buah Stroberi (*Fragaria ananassa*)," vol. 34, no. 2, pp. 255–263, 2023.
- [2] R. Y. Putri, K. Siregar, and D. Devianti, "Pertumbuhan Tanaman Stroberi (*Fragaria* sp.) Secara Hidroponik di Dataran Rendah pada Berbagai Nilai EC (Electrical Conductivity)," *J. Ilm. Mhs. Pertan.*, vol. 5, no. 1, pp. 481–490, 2020, doi: 10.17969/jimfp.v5i1.13728.
- [3] A. V. Efrilla, S. B. Sulisty, K. Wijaya, P. H. Kuncoro, and A. Sudarmaji, "Klasifikasi Penyakit Pada Daun Stroberi Menggunakan K-Means Clustering dan Jaringan Syaraf Tiruan," *J. Keteknikan Pertan. Trop. dan Biosist.*, vol. 8, no. 2, pp. 161–170, 2020, doi: 10.21776/ub.jkptb.2020.008.02.06.
- [4] BPS, "Produksi Tanaman Buah-buahan, 2021-2023," *Badan Pusat Statistik Indonesia*, 2023. <https://www.bps.go.id/id/statistics-table/2/NjIjMg==/produksi-tanaman-buah-buahan.html>
- [5] Q. N. Azizah, "Klasifikasi Penyakit Daun Jagung Menggunakan Metode *Convolutional Neural Network* AlexNet," *sudo J. Tek. Inform.*, vol. 2, no. 1, pp. 28–33, 2023, doi: 10.56211/sudo.v2i1.227.
- [6] U. Ungkawa and G. Al Hakim, "Klasifikasi Warna pada Kematangan Buah Kopi Kuning menggunakan Metode CNN Inception V3," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 11, no. 3, p. 731, 2023, doi: 10.26760/elkomika.v11i3.731.
- [7] M. A. Amrullah and M. I. Irawan, "Implementasi Jaringan Saraf Konvolusional dengan Inception-V3 untuk Deteksi Katarak Menggunakan Gambar Digital Funduskopi," *J. Sains dan Seni ITS*, vol. 12, no. 1, 2023, doi: 10.12962/j23373520.v12i1.106807.
- [8] M. Baihaqy, A. T. Wibowo, and D. Q. Utama, "Klasifikasi Tanaman Anggrek jenis *Phalaenopsis* berdasarkan Citra Labellum Bunga Menggunakan Metode Convolutinal Neural Network (CNN)," *Telkom Univ.*, vol. 9, no. 3, pp. 1942–1951, 2022.
- [9] G. Y. Christiawan, R. A. Putra, A. Sulaiman, E. Poerbaningtyas, and S. W. Putri Listio, "Penerapan Metode *Convolutional Neural Network* (CNN) Dalam Mengklasifikasikan Penyakit Daun Tanaman Padi," *J-Intech*, vol. 11, no. 2, pp. 294–306, 2023, doi: 10.32664/j-intech.v11i2.1006.
- [10] Y. A. Suwitonon and F. J. Kaunang, "Implementasi Algoritma *Convolutional Neural Network* (CNN) Untuk Klasifikasi Daun Dengan Metode Data Mining SEMMA Menggunakan Keras," *J. Komtika (Komputasi dan Inform.)*, vol. 6, no. 2, pp. 109–121, 2022, doi: 10.31603/komtika.v6i2.8054.
- [11] E. Andreas and W. Widhiarso, "Klasifikasi Penyakit Mata Katarak Menggunakan *Convolutional Neural Network* Dengan Arsitektur Inception V3," *MDP Student Conf.*, vol. 2, no. 1, pp. 107–113, 2023, doi: 10.35957/mdp-sc.v2i1.3660.
- [12] S. Alden and B. N. Sari, "Implementasi Algoritma CNN Untuk Pemilahan Jenis Sampah Berbasis Android Dengan Metode CRISP-DM," *J. Inform.*, vol. 10, no. 1, pp. 62–71, 2023, doi: 10.31294/inf.v10i1.14985.
- [13] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying CRISP-DM process model," *Procedia Comput. Sci.*, vol. 181, no. 2019, pp. 526–534, 2021, doi: 10.1016/j.procs.2021.01.199.
- [14] Y. Yudiana, A. Yulia Agustina, and dan Nur Khofifah, "Prediksi Customer Churn Menggunakan Metode CRISP-DM Pada Industri Telekomunikasi Sebagai Implementasi Mempertahankan Pelanggan," *IJIEB Indones. J. Islam. Econ. Bus.*, vol. 8, no. 1, pp. 1–20, 2023, [Online]. Available: <http://ejournal.lp2m.uinjambi.ac.id/ojp/index.php/ijoi> eb