

## **Analisis *Quality of Service* pada Rancang Bangun Pendeteksi Penyakit Tanaman Tomat Berbasis CNN dan Aplikasi Telegram**

### ***Quality of Service Analysis on Tomato Plant Disease Detection Design Based on CNN and Telegram Application***

SILVIA DWI YANTI<sup>1</sup>, YURIS MULYA SAPUTRA<sup>1\*</sup>

#### **Abstrak**

Peningkatan tanaman tomat terjadi secara konsisten dari tahun ke tahun dan berkontribusi signifikan terhadap perekonomian nasional. Peningkatan kepadatan tanaman dan perluasan area panen dapat menciptakan kondisi yang ideal untuk penyebaran penyakit pada tanaman tomat, yang berpotensi mengancam hasil panen. Penelitian ini merancang model *machine learning* menggunakan algoritme *Convolutional Neural Networks* (CNN) untuk mendeteksi penyakit tanaman tomat dengan tingkat akurasi yang diperoleh mencapai 96%. Tingkat *loss* dari model tergolong rendah yaitu sekitar 13%, membuktikan bahwa prediksi model cukup dekat dengan kondisi sebenarnya. Hasil ini menunjukkan bahwa kinerja model efektif untuk mencegah penyebaran penyakit tanaman tomat dengan membantu mengidentifikasi penyakit lebih awal. Model *machine learning* diimplementasikan melalui Bot Telegram sebagai antarmuka pengguna, yang tidak hanya efektif dalam memberikan informasi deteksi penyakit tanaman tomat, tetapi juga memastikan informasi tersampaikan dengan efisien dan tepat. Analisis *Quality of Service* (QoS) dilakukan terhadap komunikasi antara pengguna dan server Telegram dengan mempertimbangkan parameter *throughput*, *delay*, dan *packet delivery*. Nilai QoS secara keseluruhan adalah berindeks 3 kategori “Memuaskan” sesuai standarisasi versi TIPHON. Nilai QoS tersebut didapatkan dari nilai parameter *throughput* dengan indeks 4 kategori “Sangat Bagus”, nilai parameter *packet delivery* dengan indeks 4 kategori “Sangat Bagus”, serta nilai parameter *delay* dengan indeks 4 kategori “Sangat Bagus”.

Kata Kunci: Bot telegram, *convolutional neural network*, *quality of service*, tanaman tomat.

#### **Abstract**

*The increase in tomato cultivation has been consistent over the years and significantly contributes to the national economy. The increase in plant density and the expansion of harvesting areas can create ideal conditions for the spread of diseases in tomato plants, potentially threatening crop yields. This study presents a machine learning model using Convolutional Neural Networks (CNN) algorithms to detect tomato plant diseases, achieving an accuracy rate of 96%. The model demonstrates a relatively low loss rate of approximately 13%, indicating that the predictions closely align with actual conditions. These results indicate that the model's performance is effective in preventing the spread of tomato plant diseases by helping to identify diseases earlier. The machine learning model is implemented through a Telegram Bot as a user interface, which is not only effective in providing information on tomato plant disease detection, but also ensures that the information is delivered efficiently and accurately. A Quality of Service (QoS) analysis was conducted on the communication between users and the Telegram server, considering parameters such as throughput, delay, and packet delivery. The overall QoS score is indexed at 3 in the “Satisfactory” category according to TIPHON standards. This QoS score is derived from the throughput parameter with an index of 4 in the “Very Good” category, the packet delivery parameter with an index of 4 in the “Very Good” category, and the delay parameter with an index of 4 in the “Very Good” category.*

*Keywords: Convolutional neural network, quality of service, telegram bot, tomato plants.*

<sup>1</sup> Teknologi Rekayasa Internet, Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada, Yogyakarta, 55281

\* Penulis korespondensi: Surel: [ym.saputra@ugm.ac.id](mailto:ym.saputra@ugm.ac.id)

## PENDAHULUAN

Tomat (*Solanum lycopersicum*) merupakan tanaman sayuran yang terkenal dan banyak dikonsumsi oleh masyarakat Indonesia. Komoditas pertanian pada subsektor hortikultura ini, mengalami peningkatan dalam periode lima tahun terakhir, dari 2018 hingga 2022. Pertumbuhan mencolok tanaman tomat di Indonesia terlihat dari peningkatan produksi dan perluasan area panen. Basis Data Statistik Pertanian (BDSP) menunjukkan kenaikan rata-rata produksi tomat sebesar 4.575% per tahun dan kenaikan rata-rata luas area panen sebesar 4.015% per tahun (Kementerian Pertanian Republik Indonesia, 2022). Peningkatan ini secara konsisten terjadi dari tahun ke tahun dan berkontribusi signifikan terhadap perekonomian nasional, sejalan dengan tumbuhnya permintaan pasar terhadap tomat.

Ketika terdapat permintaan tinggi terhadap produk pertanian, mengakibatkan kenaikan harga di tingkat petani, yang pada akhirnya meningkatkan penghasilan para petani. Tanaman sayuran merupakan subsektor yang memiliki nilai ekonomi signifikan, berperan penting dalam mendukung ekonomi nasional, dan dapat menjadi sumber penghasilan utama bagi masyarakat (Azzura *et al.* 2017). Kenaikan produksi tanaman tomat ini menandakan ekspansi industri dan permintaan yang tinggi, serta menunjukkan bahwa petani semakin banyak mengalokasikan lahan untuk budidaya tomat. Hal ini memberikan dampak yang menguntungkan bagi sektor ekonomi dan keuangan. Pengalokasian lebih banyak lahan pertanian dengan penerapan praktik bertani yang efektif, dapat meningkatkan taraf hidup para petani (H Juliansyah dan A Riyono, 2018). Selain ini, hal ini dapat menciptakan peluang bisnis yang lebih besar dan potensi untuk peningkatan pendapatan.

Namun, di sisi lain, terdapat dampak negatif yang perlu diperhatikan. Kerapatan jarak tanam dan kepadatan populasi tanaman yang tinggi dapat mengakibatkan peningkatan kelembaban di sekitar area tanaman, sehingga sangat mudah terjadi penyebaran penyakit (A Asmaliyah dan T Rostiwati, 2015). Peningkatan kepadatan tanaman dan perluasan area panen dapat menciptakan kondisi yang ideal untuk penyebaran penyakit pada tanaman tomat, yang berpotensi mengancam hasil panen dan keberlanjutan industri pertanian ini. Penyakit pada tanaman tomat mula-mula dapat diketahui dari bercak yang muncul pada daun. Gejala ini bisa dilihat dengan mudah karena perbedaan warna, tekstur, dan karakteristik unik. Namun, Rofiq (2020) menyatakan bahwa identifikasi visual ini dapat menyebabkan kebingungan akibat kemiripan antara satu penyakit dengan penyakit lainnya, yang pada akhirnya dapat mengakibatkan diagnosis yang kurang tepat.

Dalam konteks di mana produksi dan keuntungan petani berpotensi terancam oleh penyakit tanaman, menjadi sangat penting untuk menemukan solusi yang cepat dan efisien untuk mengatasi masalah ini. Dalam upaya meningkatkan deteksi dini, khususnya mendeteksi penyakit pada tanaman tomat, penggunaan teknologi berbasis *machine learning* telah mengalami kemajuan signifikan. Penggunaan algoritme *Convolutional Neural Networks* (CNN) adalah cara paling efektif untuk membantu mengidentifikasi penyakit tanaman pada kondisi pertanian pada tahap awal (Sharma, Berwal dan Ghai, 2020).

Penelitian terkait pembuatan model *machine learning* untuk mendeteksi penyakit tanaman tomat sudah banyak dilakukan sebelumnya. Penelitian pernah dilakukan oleh Durmus *et al.* (2017), Ashok *et al.* (2020), Natbais *et al.* (2023), dan Soekarta *et al.* (2023). Kekurangan pada penelitian yang pernah dilakukan adalah belum adanya antarmuka pengguna yang tepat untuk mengakses dan memahami informasi dari model tersebut. Komunikasi pengguna dan sistem deteksi harus dapat menjalankan antarmuka yang efektif, sehingga memungkinkan pertukaran informasi terjadi dengan cepat dan tepat. Salah satu aplikasi yang dapat digunakan sebagai antarmuka sistem deteksi yang cepat dan efisien adalah Bot Telegram.

Bot Telegram merupakan teknologi cerdas yang mampu menjalankan fungsi secara otomatis. Telegram mengklaim dirinya sebagai aplikasi olah pesan yang sangat cepat, yang berfokus pada kecepatan dan keamanan. Telegram dapat digunakan di semua perangkat pada waktu yang bersamaan, dan semua pesan akan tersinkronisasi secara otomatis (Telegram 2023).

Meskipun aplikasi pendeteksi dengan Bot Telegram ini menjanjikan, tetapi faktor *Quality of Service* (QoS) dalam komunikasi antara pengguna dan *server* telegram sangat penting untuk dianalisis. Respon yang cepat dan akurat dari *server* menjadi sangat krusial karena keterlambatan atau kesalahan dalam diagnosa bisa mengakibatkan penyebaran penyakit yang lebih luas dan meningkatkan kerugian. Oleh karena itu, penting untuk mengambil langkah-langkah dalam mengatasi permasalahan ini dengan mengembangkan sebuah model *machine learning* untuk mendeteksi penyakit tanaman tomat dan mengimplementasi antarmuka pengguna melalui Bot Telegram, yang tidak hanya efektif dalam memberikan informasi deteksi penyakit tanaman tomat, tetapi juga memastikan informasi tersampaikan dengan efisien dan tepat antara pengguna dan sistem deteksi.

Penelitian terkait deteksi penyakit pada tanaman tomat dilakukan oleh Ankita *et al.* (2023). Penelitian ini merancang sistem deteksi penyakit pada tanaman tomat dengan menggunakan model *U-Net* untuk segmentasi gambar dan *convolutional network* untuk klasifikasi, dan membandingkan beberapa algoritme. Hasil penelitian menunjukkan bahwa kombinasi teknik segmentasi dan CNN sangat berhasil dalam mendeteksi penyakit tanaman tomat secara otomatis jika dibandingkan dengan algoritme *InceptionV3*, *Modified MobileNet*, dan *ResNet50*. Pada tahun 2023, sebuah penelitian yang membandingkan algoritme CNN dengan SVM juga dilakukan (Fiviana *et al.* 2023). Penelitian tersebut merancang sistem deteksi penyakit pada tanaman jagung melalui citra daun secara otomatis dan cepat. Hasilnya menunjukkan bahwa metode CNN menghasilkan akurasi tertinggi, yakni 98%, sementara SVM hanya mencapai 87%. Chaganti *et al.* (2020) juga melakukan perbandingan antara algoritme CNN dan SVM. Hasilnya menunjukkan bahwa CNN mencapai akurasi yang lebih tinggi, yaitu 93.57%, dibandingkan dengan SVM yang hanya mencapai akurasi 82%.

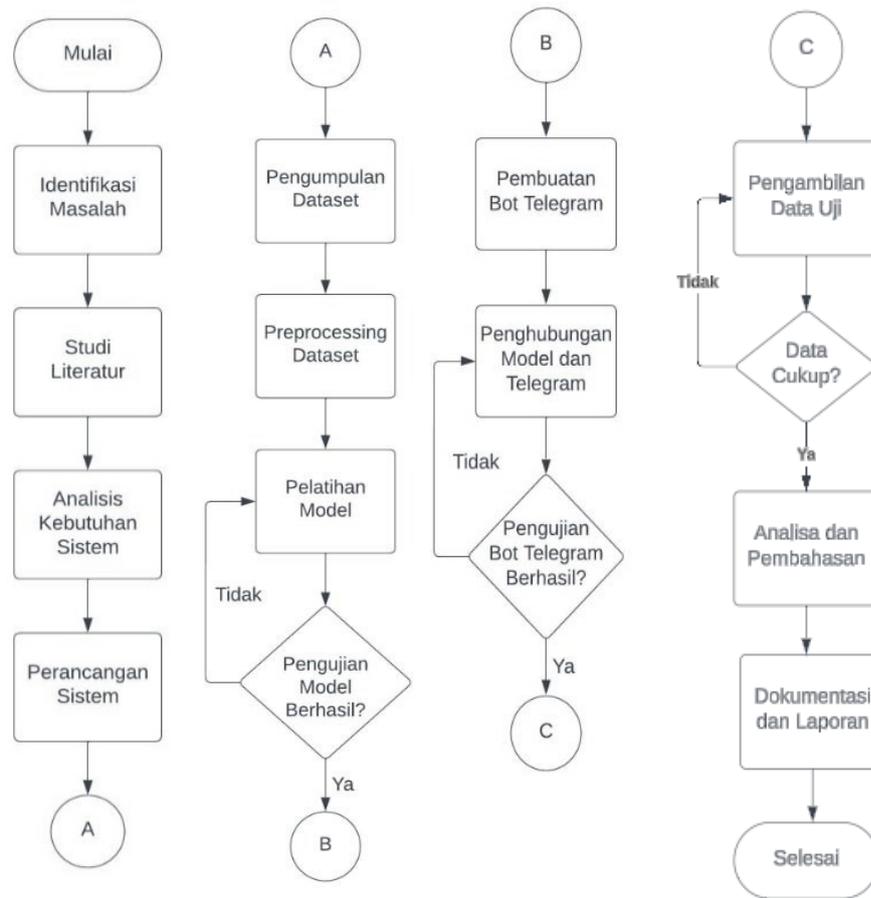
Pada penelitian ini, dilakukan deteksi terhadap penyakit tanaman tomat berdasarkan citra daun, dengan menonjolkan antarmuka pengguna yang inovatif dalam bentuk Bot Telegram. Hal ini memungkinkan akses dan pemahaman informasi dari model CNN menjadi lebih mudah bagi pengguna. Integrasi antarmuka pengguna berbasis Bot Telegram juga membuka potensi untuk membangun komunitas pengguna yang berbagi informasi tentang penyakit tanaman dan solusi deteksi yang ditemukan. Dengan demikian, penggunaan Telegram tidak hanya efektif dalam memberikan informasi deteksi penyakit tanaman tomat, tetapi juga memiliki potensi untuk meningkatkan skala dan aksesibilitas solusi deteksi tersebut. Selain itu, penelitian ini juga mengintegrasikan analisis QoS untuk mengatasi kendala komunikasi, memastikan bahwa Bot Telegram dapat menyampaikan informasi dengan efisien dan akurat, serta memberikan respons yang cepat dan tepat.

## METODE

Metode penelitian ini diilustrasikan pada Gambar 1 sebagai representasi visual tahapan yang dilalui dalam penelitian. Diagram menggambarkan proses pengembangan dan evaluasi sistem yang terintegrasi dengan Bot Telegram, dimulai dengan identifikasi masalah, diikuti oleh pengumpulan dan pemrosesan data, termasuk membagi menjadi 80% data latih dan 20% data uji pada tahap praproses data, kemudian perancangan dan pelatihan model *machine learning*. Setelah model dilatih, dilakukan integrasi dan pengujian model dengan Bot Telegram, menyesuaikan dan mengumpulkan data untuk proses analisis, hingga akhirnya penelitian ini selesai dengan dokumentasi.

Pada tahap analisis untuk memperoleh hasil akhir Qos, dilakukan dengan menjumlahkan semua indeks yang diperoleh berdasarkan berbagai parameter yang digunakan dalam perhitungan dengan rumus pada Persamaan 1.

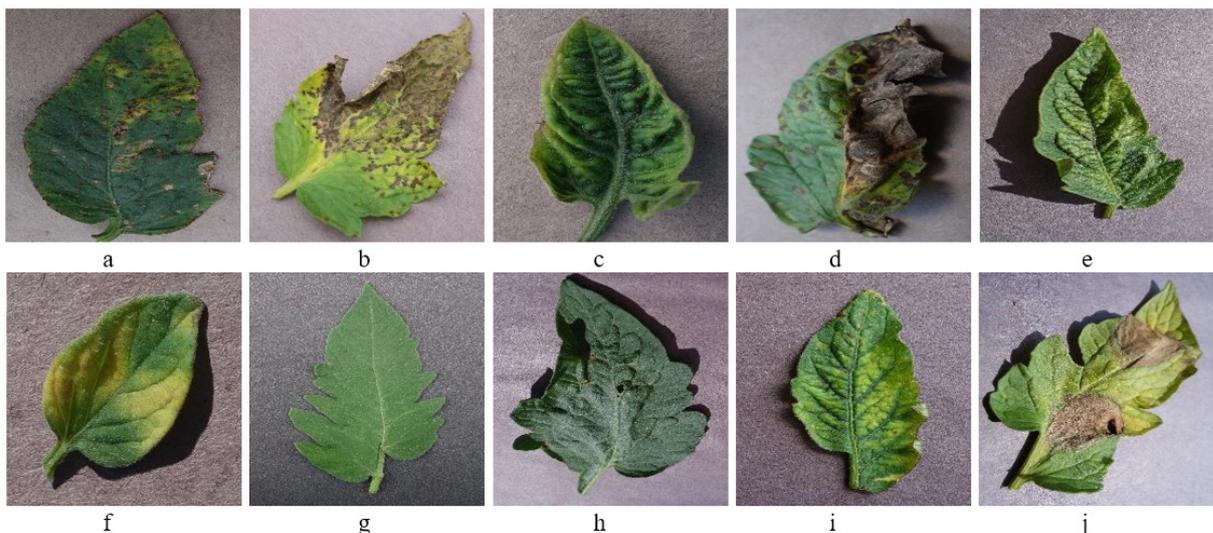
$$QoS = \frac{\sum \text{Indeks Parameter}}{\text{Jumlah Parameter}} \quad (1)$$



Gambar 1 Diagram alir sistem deteksi penyakit tanaman tomat

### Data Gambar (Bahan Penelitian)

Penelitian ini membangun sebuah sistem aplikasi pendeteksi penyakit tanaman tomat berdasarkan gambar daun. Data gambar sebagai bahan penelitian menggunakan *dataset* penyakit daun tanaman tomat yang sudah diberi label dan siap digunakan berdasarkan 9 kelas penyakit dan 1 kelas sehat, seperti terlihat pada Gambar 2. *Dataset* tersebut yakni *PlantVillage-Dataset* (spMohanty 2018) yang dapat diakses melalui [Github](#) dan *PlantifyDr Dataset* (Alex Lavaee 2020) yang dapat diakses melalui portal [Kaggle](#).



Gambar 2 Kelas tanaman tomat; (a) *Bacterial Spot*, (b) *Septoria Leaf Spot*, (c) *Yellow Leaf Curl Virus*, (d) *Early Blight*, (e) *Two-Spotted Spider Mites*, (f) *Leaf Mold*, (g) *Healthy*, (h) *Target Spot*, (i) *Mosaic Virus*, (j) *Late Blight*.

## Perancangan Model dari Sistem

*Dataset* yang tersedia telah dilengkapi label dan siap digunakan. Namun, terdapat ketidakseimbangan dalam jumlah data antar kelasnya, sehingga tahap *preprocessing* yang dilakukan adalah menyeimbangkan jumlah data di tiap kelas. Tabel 1 merupakan hasil jumlah data yang telah seimbang.

Penelitian ini menggunakan Google Colab yang sudah ter-*install* bahasa Python untuk membangun model. Persiapan yang dilakukan adalah men-*download dataset* yang sebelumnya telah disimpan pada [Google Drive](#), membuat direktori *dataset train* dan *test*, lalu *splitting data*. Sebelum memulai proses *training*, model dibangun menggunakan lapisan CNN yang bekerja dengan mengekstrak fitur dari *input* melalui operasi konvolusi, yang secara bertahap mengurangi dimensi data, namun tetap mempertahankan karakteristik yang penting. Hal ini membuat model lebih efisien dan akurat dalam memprediksi atau mengklasifikasi data. Model CNN dilatih untuk klasifikasi gambar dengan menggunakan berbagai teknik, seperti *augmentasi gambar*, *dropout*, dan *early stopping* untuk meningkatkan performa dan mencegah *overfitting*.

Pada perancangan dan pelatihan model CNN, digunakan beberapa *hyperparameter* penting untuk memastikan kinerja dan *reproducibility* yang optimal. Model ini terdiri dari empat lapisan konvolusi dengan aktivasi ReLU dan *max pooling*, serta lima lapisan *dense* dengan *dropout* untuk mencegah *overfitting*. Ukuran *batch* yang digunakan adalah 32, dengan jumlah lapisan konvolusi sebanyak empat. *Learning rate* yang diterapkan adalah 0.001. *Dropout rate* yang digunakan bervariasi antara 0.1, 0.2, 0.3, dan 0.4. Algoritme optimasi yang digunakan adalah Adam. Proses *tuning* dilakukan dengan mencoba berbagai kombinasi *hyperparameter* pada data validasi untuk menemukan konfigurasi yang memberikan kinerja terbaik. Percobaan ini melibatkan penyesuaian ukuran *batch*, *learning rate*, *dropout rate*, dan jumlah lapisan untuk mencapai keseimbangan optimal antara akurasi dan generalisasi model. Evaluasi model dengan *confusion matrix* dilakukan untuk melihat akurasi dari model yang dibangun.

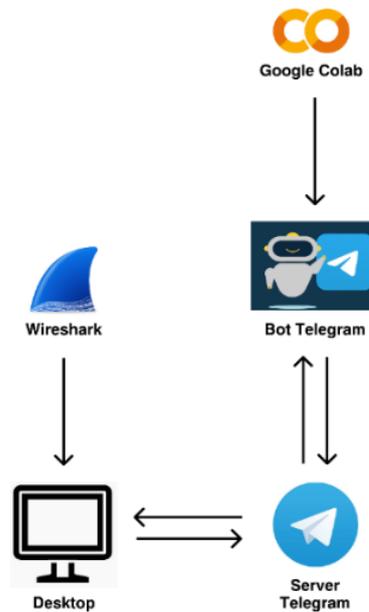
## Perancangan Topologi Pengujian Sistem

Pengujian sistem dilakukan menggunakan Wireshark untuk menangkap paket data yang dikirimkan pengguna (*desktop*) melalui telegram. Ketika pengguna berinteraksi dengan Bot Telegram, paket-paket data yang dikirimkan dari desktop akan melewati *server* telegram dan akan ditindaklanjuti oleh model *machine learning* yang dijalankan di Google Colab. Semua paket data yang bergerak antara *desktop* dan *server* telegram ditangkap oleh Wireshark untuk dilakukan analisis lebih lanjut. Proses ini dapat dilihat pada Gambar 3.

Data pengujian terdiri dari gambar yang tidak digunakan selama pelatihan, memiliki distribusi dan karakteristik yang mirip dengan data pelatihan, dan dilabeli dengan benar. Evaluasi terhadap keluaran sistem dilakukan menggunakan metrik seperti akurasi. Selain itu, dilakukan pengukuran *loss* pada data pengujian untuk menilai seberapa besar kesalahan prediksi model. Hasil pengujian dianggap cukup baik jika model menunjukkan akurasi tinggi dan metrik evaluasi yang seimbang, yang menunjukkan bahwa model dapat diandalkan untuk digunakan dalam aplikasi dunia nyata.

Tabel 1 Jumlah data tiap kelas tanaman tomat

Jenis Penyakit	Jumlah Data
Healthy	2876
Bacterial Spot	2852
Early Blight	2974
Late Blight	2973
Leaf Mold	2952
Mosaic Virus	2655
Septoria Leaf Spot	2960
Target Spot	2932
Two Spotted Spider Mite	2888
Yellow Leaf Curl Virus	2914



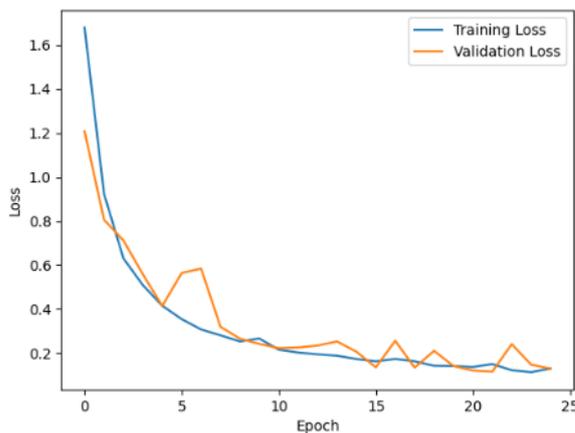
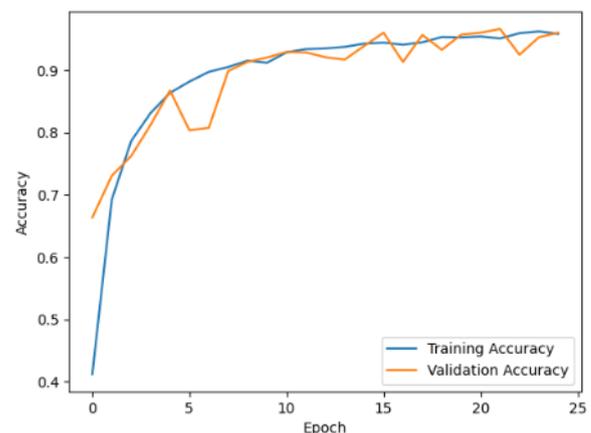
Gambar 3 Struktur topologi pengujian menggunakan Wireshark

## HASIL DAN PEMBAHASAN

### Hasil Model *Machine Learning*

Penelitian yang dilakukan melatih model *machine learning* dengan algoritme CNN dengan jumlah *dataset* penyakit tanaman tomat yang berbeda. Perbandingan dilakukan untuk mendapatkan hasil akurasi terbaik. *Dataset* pertama berjumlah 28976 gambar dan *dataset* kedua berjumlah 5000 gambar yang dibagi ke dalam 10 kelas. Hasil akurasi model pertama sebesar 96% dan *loss* 13%, sedangkan model kedua memperoleh hasil akurasi sebesar 86% dan *loss* 45%. Model pertama seperti pada Gambar 4 dan Gambar 5 merupakan model terbaik yang dipilih untuk digunakan pada pengintegrasian dengan Bot Telegram dan dilakukan analisis QoS, dikarenakan menghasilkan metrik dengan nilai terbaik. Akurasi yang tinggi (96%) menunjukkan bahwa model ini sangat efektif dalam mengidentifikasi penyakit tanaman tomat. Sedangkan *loss* atau kerugian yang rendah (13%) menunjukkan bahwa rata-rata, prediksi model cukup dekat dengan kondisi sebenarnya. Kinerja model ini sangat baik sehingga dapat membantu secara signifikan dalam mengurangi kerugian akibat penyakit tanaman tomat.

Selanjutnya, dilakukan evaluasi model untuk mengukur dan membandingkan kinerja model yang telah dilatih menggunakan metode *confusion matrix*. Hasil evaluasi menunjukkan *accuracy* sebesar 97%, yang berarti model berhasil memprediksi 97% kasus dengan benar. Selain itu, diperoleh nilai *loss* sebesar 8% yang menunjukkan bahwa model mampu menyesuaikan diri terhadap data pelatihan dengan baik, sehingga tingkat kesalahan kecil dalam memprediksi.

Gambar 4 Grafik *training and validation loss*Gambar 5 Grafik *training and validation accuracy*

Tabel 2 *confusion matrix*

		Predicted									
		0	1	2	3	4	5	6	7	8	9
Actual	0	2120	1	0	0	0	0	0	6	0	0
	1	0	986	3	2	1	1	6	1	0	0
	2	0	1	1590	0	0	0	0	0	0	0
	3	7	43	6	1826	14	0	5	8	0	0
	4	1	0	0	0	945	0	2	3	0	1
	5	0	0	0	0	0	370	1	0	2	0
	6	8	3	0	0	12	1	1745	2	0	0
	7	0	0	4	0	1	0	1	1393	5	0
	8	0	0	22	0	2	2	0	52	1598	0
	9	231	12	0	8	2	0	0	10	31	5063

Tabel 2 menunjukkan hasil visualisasi *confusion matrix* dari evaluasi model yang berisi angka-angka, yang merepresentasikan klasifikasi yang dilakukan oleh model. *Confusion matrix* ini menunjukkan gambaran tentang kinerja model dalam melakukan klasifikasi data ke dalam kategori yang benar atau salah. Matriks ini menunjukkan hubungan antara kelas sebenarnya (*Aktual*) dan kelas yang diprediksi oleh model (*Predicted*), dengan nilai-nilai diagonal mewakili jumlah prediksi yang akurat. Nilai di luar diagonal menandakan kesalahan prediksi.

### Hasil Pengujian Model *Machine Learning*

Dari model yang telah dilatih, dilakukan *testing* lanjutan untuk memastikan bahwa model telah sesuai. Proses ini dilakukan menggunakan Google Colab untuk memprediksi penyakit daun tomat dengan memasukkan gambar daun tomat sebagai *input*, lalu hasil keluaran sebagai *output* prediksi. Dalam hasil prediksi tersebut, setiap kelas penyakit ditampilkan dengan probabilitasnya masing-masing, sehingga tingkat akurasi dapat terlihat dengan jelas.

Pengujian juga dilakukan pada *data real* yang diperoleh dari data kunjungan ke kebun tomat dan pengambilan foto secara manual. Dari hasil pengujian, didapatkan sebesar 87% daun sehat, dan 3% kelas “*Tomato Bacterial spot*”, 1% kelas “*Tomato Early blight*”, 1% kelas “*Late blight*”, 2% kelas “*Tomato Leaf Mold*”, 1% kelas “*Tomato Mosaic virus*”, dan 5% kelas “*Tomato Septoria leaf spot*”, sebagaimana terlihat pada Gambar 6.

```

print("Total images processed:", total_images)
print("Class distribution:")
for class_index, (class_name, count) in class_counts.items():
    percentage = (count / total_images) * 100 if total_images > 0 else 0
    print(f"Class '{class_name}': {count} images ({percentage:.2f}%)")

Total images processed: 100
Class distribution:
Class 'Tomato Bacterial spot': 3 images (3.00%)
Class 'Tomato Early blight': 1 images (1.00%)
Class 'Tomato Healthy': 87 images (87.00%)
Class 'Tomato Late blight': 1 images (1.00%)
Class 'Tomato Leaf Mold': 2 images (2.00%)
Class 'Tomato Mosaic virus': 1 images (1.00%)
Class 'Tomato Septoria leaf spot': 5 images (5.00%)
Class 'Tomato Target Spot': 0 images (0.00%)
Class 'Tomato Two Spotted Spider Mite': 0 images (0.00%)
Class 'Tomato Yellow Leaf Curl Virus': 0 images (0.00%)

```

Gambar 6 Hasil kelas distribusi pada *data real*

### Hasil Komputasi Model *Machine Learning*

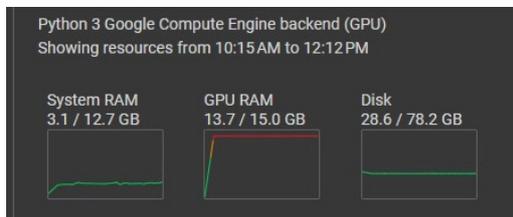
Google Colab dapat menampilkan statistik penggunaan sumber daya, seperti GPU, memori, dan *disk* selama sesi colab berjalan. Hal ini bertujuan untuk memonitor hasil beban komputasi dari sisi *server* pada Google Colab. Hasil komputasi model *machine learning* terlihat pada Gambar 8 yang menunjukkan sistem RAM digunakan sekitar 24% dari total RAM yang tersedia. GPU RAM (13.7/15 GB) menunjukkan penggunaan GPU RAM yang sangat tinggi, mencapai 91% dari kapasitas total. Hal ini mengindikasikan bahwa model sangat efisien dalam menggunakan sumber daya GPU, yang bagus untuk tugas-tugas berat, serta dapat mempertimbangkan pengurangan ukuran *batch* untuk meningkatkan optimisasi model *machine*

*learning*. Penggunaan *disk* sebanyak 28.6/78.2 GB menunjukkan masih memiliki cukup banyak ruang *disk* yang tersisa. Monitoring dan optimalisasi penggunaan sumber daya dilakukan untuk memastikan bahwa sesi Colab berjalan dengan efisien dan dapat menghindari potensi masalah kinerja atau kegagalan karena kelebihan beban.

Waktu *runtime* yang diperoleh selama sesi Colab berjalan adalah sekitar 7029.2 detik seperti terlihat pada Gambar 9. Lama proses model *machine learning* dalam menyelesaikan tugasnya dipengaruhi oleh sumber daya sistem (RAM, GPU, dan *disk*). Hasil menunjukkan bahwa penggunaan GPU yang intensif telah mempercepat proses untuk meningkatkan performa model *machine learning*.

### Hasil Pengujian Bot Telegram

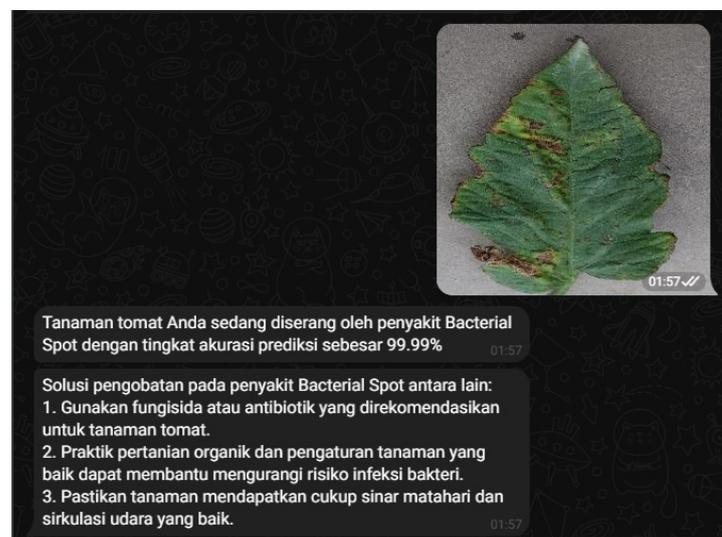
Setelah Bot Telegram berhasil dibuat, dilakukan pengujian dengan mengirimkan pesan sesuai format yang diminta. Bot Telegram ini akan mengeluarkan hasil berupa identifikasi penyakit yang dialami oleh tanaman tomat berdasarkan gambar daun yang dikirim pengguna beserta solusi pengobatan. Gambar 10 merupakan salah satu contoh keluaran hasil Bot Telegram, yang memprediksi tanaman tomat diserang oleh penyakit *Bacterial Spot*.



Gambar 7 Hasil komputasi model

```
6 end_time = time.time()
7 print(f"Total runtime: {end_time - start_time} seconds")
8
Total runtime: 7029.200933933258 seconds
```

Gambar 8 Hasil *runtime*



Gambar 9 Hasil keluaran penyakit *Bacterial Spot*

### Hasil Pengujian Parameter *Throughput*

Pengujian parameter *throughput* pada komunikasi antara pengguna dan *server* Telegram dilakukan sebanyak 5 kali dalam 3 waktu, yaitu pagi, siang, dan malam hari. Setiap pengujian dilakukan dengan durasi komunikasi  $\pm 4$  menit untuk mengumpulkan data. Pengujian ini hanya menggunakan satu *provider* internet yaitu 3 (Tri).

Tabel 3 menunjukkan hasil pengujian parameter *throughput* dalam satuan bps yang diperoleh dengan mengalikan 8 pada hasil bytes/s. Sesuai dengan standarisasi versi TIPHON, diperoleh rata-rata *throughput* pada waktu yang berbeda: 297.4 kbps di pagi hari, 282.2 kbps di siang hari, dan 301.6 kbps di malam hari. Semuanya mendapatkan indeks 4 dalam kategori

“Sangat Bagus”. Variasi nilai *throughput* selama pengujian ini relatif kecil, menunjukkan bahwa waktu dalam sehari (pagi, siang, atau malam) tidak memberikan dampak signifikan terhadap kualitas komunikasi, namun hasil *throughput* terbaik adalah pada malam hari. Nilai rata-rata *throughput* secara keseluruhan pada pengujian ini adalah 293.7 kbps dengan indeks 4 kategori “Sangat Bagus”. Gambar 10 menunjukkan hasil pengujian parameter *throughput*.

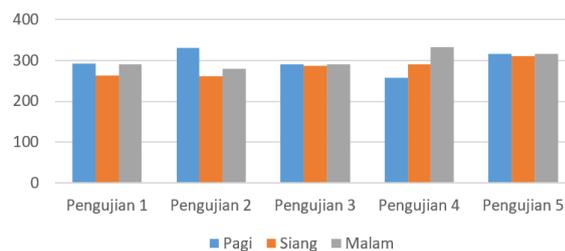
### Hasil Pengujian Parameter *Delay*

Pengujian parameter *delay* pada komunikasi antara pengguna dan *server* Telegram dilakukan sebanyak 5 kali dalam 3 waktu, yaitu pagi, siang, dan malam hari. Setiap pengujian dilakukan dengan durasi komunikasi  $\pm 4$  menit untuk mengumpulkan data. Pengujian ini hanya menggunakan satu *provider* internet yaitu 3 (Tri).

Ketika data dari Wireshark diekspor ke format CSV, berbagai jenis informasi terkait dengan lalu lintas jaringan ditampilkan. Terdapat beberapa kolom yang menampilkan informasi pada komunikasi yang terjadi antara pengguna dengan *server* Telegram yang telah disaring selama proses *capturing*. Kolom “*Time*” menunjukkan waktu selama komunikasi berlangsung. Kolom “*Source*” dan “*Destination*” mencakup *ip address* sumber dan tujuan untuk setiap paket yang dikirim. Kolom “*Protocol*” merupakan jenis protokol yang digunakan setiap paket data. Kolom “*Length*” berisi ukuran setiap paket dalam *byte*. Dan kolom “*Info*” mencakup ringkasan atau deskripsi dari isi paket. *Delay* diperoleh dengan perhitungan waktu yang dibutuhkan paket data hingga sampai ke tujuan penerimaan paket data.

Tabel 3 Hasil pengujian parameter *throughput*

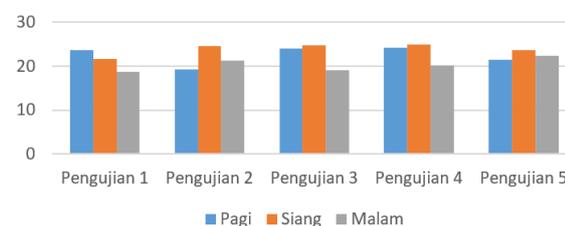
Pengujian	<i>Throughput</i> (bits/s)		
	Pagi	Siang	Malam
1	293k	263k	290k
2	331k	261k	280k
3	290k	286k	290k
4	257k	290k	332k
5	316k	311k	316k
Jumlah	1487k	1411k	1508k
Rata-rata	297.4k	282.2k	301.6k



Gambar 10 Grafik hasil pengujian parameter *throughput*

Tabel 4 Hasil pengujian parameter *delay*

Pengujian	<i>Delay</i> (ms)		
	Pagi	Siang	Malam
1	23.6	21.6	18.7
2	19.3	24.6	21.3
3	24	24.7	19
4	24.2	24.9	20.1
5	21.4	23.6	22.4
Jumlah	112.5	119.4	101.5
Rata-rata	22.5	23.88	20.3



Gambar 11 Grafik hasil pengujian parameter *delay*

Tabel 4 menunjukkan hasil pengujian parameter *delay*. Sesuai dengan standarisasi versi TIPHON, diperoleh rata-rata *delay* pada waktu yang berbeda: 22.5 ms di pagi hari, 23.88 ms di siang hari, dan 20.3 ms di malam hari. Semuanya mendapatkan indeks 4 dalam kategori “Sangat Bagus”. Variasi nilai *delay* selama pengujian ini relatif kecil, menunjukkan bahwa waktu dalam sehari (pagi, siang, atau malam) tidak memberikan dampak signifikan terhadap kualitas komunikasi, namun hasil *delay* terkecil adalah pada malam hari. Nilai rata-rata *delay* secara keseluruhan pada pengujian ini adalah 22.2 ms dengan indeks 4 kategori “Sangat Bagus”. Gambar 11 menunjukkan hasil pengujian parameter *delay*.

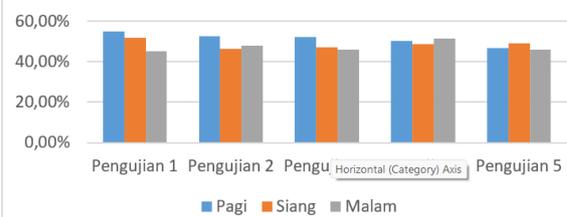
### Hasil Pengujian Parameter *Packet Loss*

Pengujian parameter *packet loss* pada komunikasi antara pengguna dan *server* Telegram dilakukan sebanyak 5 kali dalam 3 waktu, yaitu pagi, siang, dan malam hari. Setiap pengujian dilakukan dengan durasi komunikasi  $\pm 4$  menit untuk mengumpulkan data. Pengujian ini hanya menggunakan satu *provider* internet yaitu 3 (Tri).

Tabel 5 menunjukkan hasil pengujian parameter *packet loss*. Sesuai dengan standarisasi versi TIPHON, diperoleh rata-rata *packet loss* pada waktu yang berbeda: 51.42% di pagi hari, 48.64% di siang hari, dan 47.36% di malam hari. Ketiga waktu yang berbeda ini semuanya memperoleh indeks 1 dalam kategori “Buruk” karena nilai *packet loss*  $>25$ . Hal ini menunjukkan bahwa jaringan mengalami beban dan kondisi jaringan yang tidak stabil, sehingga menghasilkan *packet loss* yang tinggi dan kualitas komunikasi buruk. Nilai rata-rata *packet loss* secara keseluruhan pada pengujian ini adalah 49.14% dengan indeks 1 kategori “Buruk”. Hal ini menunjukkan kinerja jaringan yang buruk dalam kualitas komunikasi antara pengguna dan *server* Telegram. Gambar 12 menunjukkan hasil pengujian parameter *packet loss*.

Tabel 5 Hasil pengujian parameter *packet loss*

Pengujian	<i>Packet Loss (%)</i>		
	Pagi	Siang	Malam
1	54.9	58.1	45.3
2	52.8	46.4	48
3	2.4	47	45.8
4	50.2	48.8	51.6
5	46.8	49.2	46.1
Jumlah	257.1	243.2	236.8
Rata-rata	51.42	48.64	47.36



Gambar 12 Grafik hasil pengujian parameter *packet loss*

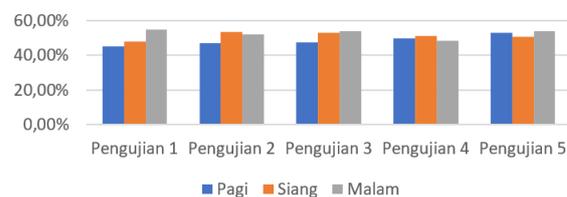
Dampak dari tingkat *packet loss* yang tinggi ini dapat menyebabkan kesalahan atau ketidakakuratan prediksi karena kualitas gambar yang mungkin rendah atau kabur, sehingga sistem sulit mengidentifikasi fitur untuk memprediksi penyakit. Gangguan dalam respons bot dan keterlambatan dalam interaksi juga dapat terjadi karena bot kesulitan memproses informasi yang diterima secara tidak lengkap, sehingga merespons dengan tepat menjadi terhambat. Namun, hal ini dapat diatasi dengan penanganan yang sesuai terhadap masalah jaringan untuk memastikan pengiriman gambar secara lengkap sebagai informasi model untuk memprediksi kelas penyakit tanaman tomat.

### Hasil Pengujian Parameter *Packet Delivery*

Pengujian parameter *packet delivery* pada komunikasi antara pengguna dan *server* Telegram dilakukan sebanyak 5 kali dalam 3 waktu, yaitu pagi, siang, dan malam hari. Setiap pengujian dilakukan dengan durasi komunikasi  $\pm 4$  menit untuk mengumpulkan data. Pengujian ini hanya menggunakan satu *provider* internet yaitu 3 (Tri).

Tabel 6 Hasil pengujian parameter *packet delivery*

Pengujian	<i>Packet Delivery (%)</i>		
	Pagi	Siang	Malam
1		48.2	
2	47.2	53.6	52
3	47.6	53	54.2
4	49.8	51.2	48.4
5	53.2	50.8	53.9
Jumlah	242.9	256.8	263.2
Rata-rata	48.58	51.36	52.64



Gambar 13 Grafik hasil pengujian parameter *packet delivery*

Tabel 6 menunjukkan hasil pengujian parameter *packet delivery*, diperoleh rata-rata pada waktu yang berbeda: 48.58% di pagi hari, 51.36% di siang hari, dan 52.64% di malam hari. 3 waktu berbeda ini, sama-sama memperoleh indeks 4 dalam kategori “Sangat Bagus” karena nilai *packet delivery* >25. Nilai rata-rata *packet delivery* secara keseluruhan pada pengujian ini adalah 50.86% dengan indeks 4 kategori “Sangat Bagus”. Hal ini menunjukkan kinerja jaringan yang cukup baik dalam kualitas komunikasi antara pengguna dan *server* Telegram. Gambar 13 menunjukkan hasil pengujian parameter *packet delivery*.

### Hasil Keseluruhan Pengujian QoS

Hasil keseluruhan pengujian QoS didapatkan dari penggabungan hasil pengujian parameter *throughput*, *delay*, dan *packet delivery* yang didapatkan, seperti terlihat pada tabel 7.

Tabel 7 Hasil pengujian parameter QoS

Parameter	Kategori	Indeks
Throughput	Sangat Bagus	4
Delay	Sangat Bagus	4
Packet delivery	Sangat Bagus	4

Pengujian QoS diperoleh dengan menjumlahkan keseluruhan indeks yang didapat dan dibagi dengan jumlah pengujian parameter yang dilakukan. Persamaan 2 menunjukkan perhitungan nilai QoS pada pengujian ini, dengan nilai sebesar 4 pada indeks “Sangat Memuaskan”.

$$QoS = \frac{4 + 4 + 4}{3} = 4 \quad (2)$$

### SIMPULAN

Berdasarkan hasil analisis QoS pada sistem aplikasi pendeteksi penyakit tanaman tomat berbasis CNN dan aplikasi Telegram, kesimpulan yang diperoleh adalah sistem yang dirancang menggunakan model CNN yang terintegrasi Bot Telegram dapat berjalan dengan baik untuk mendeteksi penyakit tanaman tomat berdasarkan gambar daun dengan tingkat akurasi mencapai 96% dan *loss* 13%. Hasil ini menunjukkan kinerja model yang efektif untuk mencegah penyebaran penyakit tanaman tomat dengan membantu mengidentifikasi penyakit lebih awal. Selanjutnya, Bot Telegram yang dibuat dapat menunjukkan hasil klasifikasi penyakit tanaman tomat berdasarkan gambar daun beserta solusi pengobatan. Nilai QoS dari komunikasi yang terjadi antara pengguna dan aplikasi Telegram dengan pengujian parameter *throughput*, *delay*, dan *packet delivery* berindeks 4 kategori “Sangat Memuaskan”, sesuai dengan standarisasi versi TIPHON. Nilai QoS tersebut didapatkan dari nilai parameter *throughput* reratanya sebesar 293.7 kbps dengan indeks 4 kategori “Sangat Bagus”, nilai parameter *packet delivery* reratanya sebesar 50.86% dengan indeks 4 kategori “Sangat Bagus”, serta nilai parameter *delay* reratanya sebesar 22.2 ms dengan indeks 4 kategori “Sangat Bagus”.

### DAFTAR PUSTAKA

- Ashok S, Kishore G, Rajesh V, Suchitra S, Sophia S, G, Pavithra B. 2020. Tomato leaf disease detection using deep learning techniques. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. 979–983. IEEE.
- Asmaliyah A, Rostiwati T. 2015. Pengaruh pengaturan jarak tanam terhadap perkembangan serangan hama dan penyakit pulai darat (*Alstonia Angustiloba*). *Jurnal Penelitian Hutan Tanaman* [Preprint].

- Azzura D, Marsudi E, Usman M. 2017. Analisis pendapatan usahatani sayur-sayuran dan faktor-faktor yang mempengaruhinya di kecamatan Darussalam kabupaten Aceh Besar. *Jurnal Ilmiah Mahasiswa Pertanian Unsyiah*. Available at: [www.jim.unsyiah.ac.id/JFP](http://www.jim.unsyiah.ac.id/JFP).
- Durmus H, Güneş EO, Kırıcı M. 2017. Disease detection on the leaves of the tomato plants by using deep learning. In *2017 6th International conference on agro-geoinformatics*. 1–5. IEEE.
- Juliansyah H, Riyono A. 2018. Pengaruh produksi, luas lahan dan tingkat pendidikan terhadap pendapatan petani karet di desa Bukit Hagu kecamatan Lhoksukon kabupaten Aceh Utara. *Jurnal Ekonomi Pertanian Unimal* [Preprint].
- Kementerian Pertanian Republik Indonesia. 2022. *Situs Basis Data Statistik Pertanian*. <https://bdsp2.pertanian.go.id/bdsp/id/home.html>.
- Lavaee A. 2020. PlantifyDr Dataset. Available at: <https://www.kaggle.com/datasets/lavaman151/plantifydr-dataset/> (Accessed: 14 December 2023).
- Natbais YH, Umbu ABS. 2023. Aplikasi deteksi penyakit pada daun tomat berbasis android menggunakan model terlatih tensorflow lite. *Teknotan: Jurnal Industri Teknologi Pertanian*. 17(2): 83–90.
- Sharma P, Berwal YPS, Ghai W. 2020. Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. *Information Processing in Agriculture*. 7(4): 566–574. Available at: <https://doi.org/10.1016/j.inpa.2019.11.001>.
- Soekarta R, Nurdjan N, Syah A. 2023. Klasifikasi penyakit tanaman tomat menggunakan metode Convolutional Neural Network (CNN). *Insect (Informatics and Security): Jurnal Teknik Informatika*. 8(2): 143–151.
- spMohanty. 2018. Dataset of diseased plant leaf images and corresponding labels. Available at: <https://github.com/spMohanty/PlantVillage-Dataset/tree/master> (Accessed: 13 December 2023).