



Identifikasi penyakit tanaman tomat melalui citra daun menggunakan DenseNet201

Okamisar¹, Regiolina Hayami², Evans Fuad

Email: ¹200401041@student.umri.ac.id, ²regiolinahayami@umri.ac.id, ³evansfuad@umri.ac.id

¹²³Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau

Diterima: 05 Agustus 2025 | Direvisi: - | Disetujui: 13 Agustus 2025
©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Muhammadiyah Riau, Indonesia

Abstrak

Penelitian ini berfokus pada implementasi algoritma *DenseNet201* untuk klasifikasi penyakit pada tanaman tomat menggunakan dataset gambar daun dari *PlantVillage*. Sektor pertanian memiliki peranan sentral dalam perekonomian Indonesia, dengan tomat sebagai salah satu tanaman hortikultura penting. Namun, produktivitas tomat seringkali terhambat oleh berbagai penyakit tanaman. Diagnosis penyakit secara akurat menjadi krusial untuk meningkatkan stabilitas produksi. Pendekatan berbasis pemrosesan citra, seperti *Convolutional Neural Network (CNN)*, telah memfasilitasi diagnosis penyakit tanaman secara efektif. Dalam penelitian ini, dataset *PlantVillage* yang terdiri dari 18.835 gambar daun tomat digunakan. Data dibagi menjadi train (10.000 gambar), validasi (7.000 gambar), dan tes (500 gambar). Model klasifikasi dibangun menggunakan arsitektur *DenseNet201* dengan beberapa modifikasi. Hasil menunjukkan bahwa model *DenseNet201* mencapai akurasi 95,20% pada data pengujian, dengan nilai F1-score keseluruhan 0,95. Dibandingkan dengan penelitian sebelumnya menggunakan *VGG16* (77,2%), *InceptionV3* (63,4%), dan *MobileNet* (63,75%), model *DenseNet201* menunjukkan peningkatan performa yang signifikan. Penelitian ini menyimpulkan bahwa *DenseNet201* sangat efektif dalam mengklasifikasikan penyakit tanaman tomat dan berpotensi untuk diimplementasikan dalam aplikasi diagnosis penyakit tanaman secara luas.

Kata kunci: Klasifikasi Penyakit Tanaman, Tomat, *DenseNet201*, *Convolutional Neural Network*, Pemrosesan Citra

Identification of tomato plant diseases through leaf images using DenseNet201

Abstract

This study focuses on implementing the *DenseNet201* algorithm for disease classification in tomato plants using leaf images from *PlantVillage* dataset. The agricultural sector plays a central role in the Indonesian economy, with tomatoes being one of the important horticultural crops. However, tomato productivity is often hindered by various plant diseases. Accurate disease diagnosis is crucial for improving production stability. Image processing-based approaches, such as *Convolutional Neural Network (CNN)*, have facilitated effective plant disease diagnosis. In this study, the *PlantVillage* dataset consisting of 18,835 tomato leaf images is utilized. The data is divided into training (10,000 images), validation (7,000 images), and test (500 images) sets. A classification model is constructed using the *DenseNet201* architecture with some modifications. The results show that the *DenseNet201* model achieves an accuracy of 95.20% on the testing data, with an overall F1-score of 0.95. Compared to previous studies using *VGG16* (77.2%), *InceptionV3* (63.4%), and *MobileNet* (63.75%), the *DenseNet201* model demonstrates a significant performance improvement. This study concludes that *DenseNet201* is highly effective in classifying tomato plant diseases and has the potential to be implemented in widespread plant disease diagnosis applications.

Keywords: Plant Disease Classification, Tomato, *DenseNet201*, *Convolutional Neural Network*, Image Processing

1. PENDAHULUAN

Pertanian memegang peranan sentral sebagai sumber pendapatan utama di berbagai negara, termasuk Indonesia. Dalam konteks Indonesia, sektor pertanian tidak hanya menjadi sumber pendapatan utama bagi sebagian penduduk, khususnya di pedesaan [1].

Solanum Lycopersicum, yang lebih dikenal sebagai tomat, merupakan tanaman hortikultura yang umum tumbuh subur di berbagai wilayah Indonesia. Keberadaan tomat bukan hanya sebagai tanaman, melainkan juga sebagai buah, menjadikan komponen penting dalam pangan dan pola konsumsi masyarakat Indonesia [2].

Menurut Badan Pusat Statistik [3], produksi tomat di Indonesia dari tahun ke tahun mengalami fluktuasi, sehingga diperlukan adanya upaya untuk meningkatkan stabilitas produksi tersebut. Banyaknya faktor yang membuat rendahnya produktivitas tomat di Indonesia yaitu salah satunya merebaknya penyakit pada tanaman tomat, hal ini dikarenakan kurangnya pengetahuan petani mengenai jenis penyakit yang dapat menyerang tanaman tomat. Saat ini, banyak petani yang masih mengandalkan pengalaman dan pengetahuan rekan-rekannya untuk mengidentifikasi dan mengatasi potensi penyakit [4].

Perkembangan dalam teknologi computer vision dan image acquisition telah memfasilitasi diagnosis penyakit tanaman melalui pendekatan pemrosesan citra. Informasi yang penting terdapat dalam gambar dan dapat diungkap melalui metode komputasional. Keakuratan sistem diagnosis ini bergantung pada kualitas gambar, yang dipengaruhi oleh sejumlah faktor seperti kondisi pencahayaan, orientasi, sensor, dan jarak [5].

Berbagai metode klasifikasi, seperti *Support Vector Machine (SVM)*, *Artificial Neural Network (ANN)*, *Convolutional Neural Network (CNN)*, *K-means clustering*, dan *Random Forest*, dapat digunakan untuk mengenali penyakit tanaman berdasarkan citra. Dari seluruh teknik tersebut, *CNN* khususnya telah mencapai pencapaian yang sangat baik dalam menangani tantangan klasifikasi pada konteks ini [6]. Keunggulan *CNN* terletak pada kemampuannya melakukan pembelajaran fitur dan klasifikasi tanpa memerlukan proses pra-pemrosesan data yang kompleks. Penting untuk di garis bawahi keberhasilan *CNN* sangat tergantung pada kualitas pelatihan, yang membutuhkan jumlah sampel atau data yang memadai serta variasi yang cukup. Hal ini mendukung kemampuan model dalam mengidentifikasi dengan efektif berbagai kondisi dan gejala penyakit tanaman [7].

Penelitian terdahulu tentang *Convolutional Neural Network (CNN)* yaitu perbandingan Akurasi, Recall, Presisi dan F1-Score pada subset berupa gambar daun tomat di dataset *PlantVillage* dengan menggunakan arsitektur *VGG16*, *InceptionV3*, dan *MobileNet*, dapat disimpulkan hasil penelitian pada arsitektur *VGG16* di dapatkan akurasi sebesar 77.2%. Pada arsitektur *InceptionV3* di dapatkan akurasi sebesar 63.4%. Pada arsitektur *MobileNet* di dapatkan akurasi sebesar 63.75% [8].

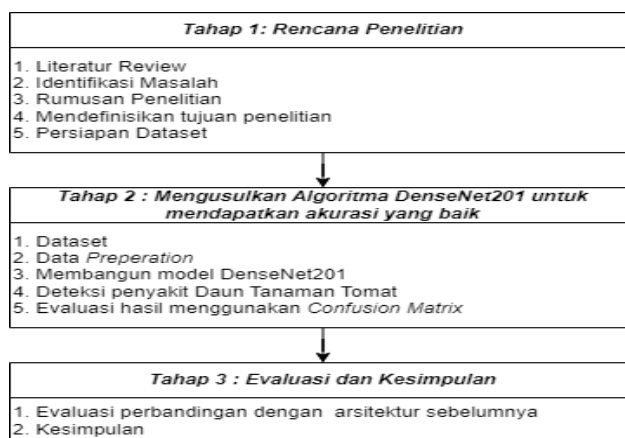
Adapun penelitian yang serupa dengan menggunakan dataset *PlantVillage* pada subset berupa gambar daun apel dengan menggunakan 10 model arsitektur *CNN* yang terdiri dari *DenseNet201*, *DenseNet169*, *InceptionV3*, *InceptionResNetV2*, *MobileNet*, *MobileNetV2*, *ResNet50*, *VGG16*, *VGG19*, dan *Xception* yang diperoleh hasil bahwa *DenseNet201* mengungguli dari model lain dengan akurasi sebesar 98,75% [7].

Kemudian penelitian lain mengembangkan tiga model deep learning yang dapat mendeteksi gulma dengan jumlah 15.336 data gambar, penyakit tanaman dengan 55.449 gambar daun dari berbagai jenis tanaman, dan hama dengan 546.183 gambar untuk hama tanaman. Penelitian ini menggunakan model arsitektur yang dibangun seperti *DenseNet201*, *MobileNet*, *InceptionV3*, *VGG16*, dan *Hyperparameter Search 2D Layer* pada data gambar pertanian, model *DenseNet201* memberikan akurasi tertinggi 99,62% dalam mendeteksi penyakit tanaman, *MobileNet* akurasi 91,85% untuk deteksi hama, dan *Hyperparameter Search 2D Layer* akurasi 71,07% untuk deteksi gulma [9].

Berdasarkan pemaparan diatas, penelitian ini akan dilakukan percobaan menggunakan arsitektur *DenseNet201* dalam memprediksi penyakit tanaman daun tomat dari dataset *PlantVillage* dan membandingkan dengan arsitektur *VGG16*, *InceptionV3*, dan *MobileNet* untuk menghasilkan tingkat Akurasi, Recall, Presisi dan F1-Score.

2. METODE PENELITIAN

Metode penelitian adalah suatu rancangan terstruktur yang disusun secara sistematis dalam melakukan penelitian yang bertujuan untuk mendapatkan hasil yang diinginkan [10]. Pada penelitian ini tahapan yang dilakukan dapat dilihat pada kerangka kerja penelitian Gambar 1.



Gambar 1 Kerangka Kerja Penelitian

2.1. Rencana Penelitian dan Persiapan Dataset

Perencanaan penelitian disusun mengikuti langkah – langkah yang harus dilalui. Dalam perencanaan penelitian, hal yang dilakukan mencakup:

2.1.1. Literatur Review

Pada tahap literatur review, dilakukan pencarian penelitian terkait studi kasus prediksi yang menggunakan algoritma DenseNet201. Sumber referensi yang dipakai yaitu jurnal – jurnal hasil penelitian terdahulu.

2.1.2. Identifikasi Masalah

Setelah melakukan tahapan pada Literatur Review bahwa diketahui perlunya pendekatan Machine Learning dengan menerapkan algoritma DenseNet201 untuk mendapatkan akurasi yang lebih baik dalam memprediksi penyakit daun tanaman tomat dan membandingkan dengan hasil akurasi pada penelitian terdahulu [11] [12].

2.1.3. Rumusan Penelitian

Pada identifikasi masalah yang dijelaskan sebelumnya, dapat ditentukan pada penelitian ini yaitu bagaimana mendapatkan hasil prediksi yang baik dan mendapatkan akurasi yang bagus.

2.1.4. Mendefinisikan Tujuan Penelitian

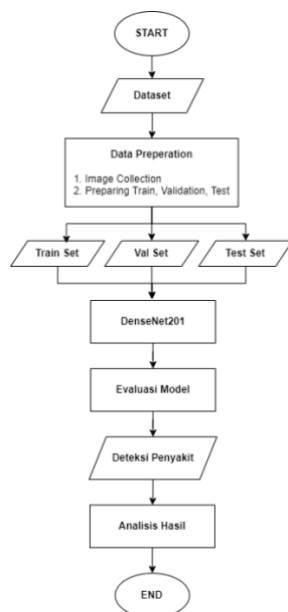
Pada penelitian ini penulis mengimplementasikan pembelajaran Machine learning dengan memprediksi penyakit tanaman daun tomat pada dataset PlantVillage. Python digunakan untuk nantinya data akan di gabungkan dan kemudian dijalankan. Penulis berharap dengan melakukan penelitian menggunakan algoritma DenseNet201 ini dapat meningkatkan hasil akurasi dan dapat memprediksi secara akurat.

2.1.5. Persiapan Dataset

Dataset yang digunakan bersumber dari plantvillage.org, ini merupakan dataset yang secara public dapat diambil pada website (https://www.tensorflow.org/datasets/catalog/plant_village). Dataset ini mencakup lebih dari 50.000 gambar dari 14 tanaman, seperti apel, blueberry, ceri, jagung, anggur, jeruk, persik, labu, kentang, raspberry, kedelai, paprika, strawberry dan tomat. Tomat dipilih sebagai bahan penelitian ini.

2.2. Mengusulkan Algoritma DenseNet201 untuk mendapatkan akurasi yang baik

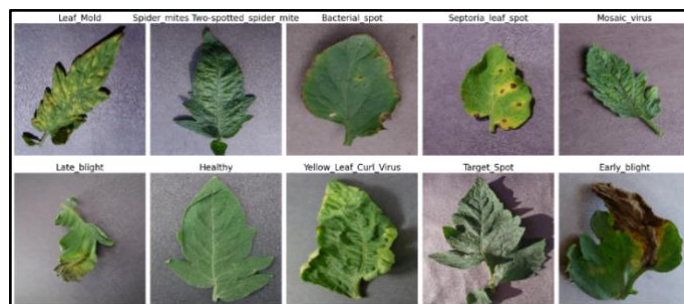
Pada tahapan alur kerja penelitian pada gambar 3.2 menjelaskan diagram alur untuk mengimplementasi algoritma DenseNet201.



Gambar 2 Alur Algoritma DenseNet201

2.2.1. Dataset

Tomat merupakan tanaman target pada penelitian ini yang di visual kan pada gambar 3.3 pada dasarnya data tomat ini memiliki sembilan jenis penyakit dan satu data sehat yang mana terdiri dari: 1. Bacterial spot, 2. Early blight, 3. Late blight, 4 Leaf mold, 5. Septoria leaf spot, 6. Spider mites: Two-spotted spider mite, 7. Target spot, 8. Tomato mosaic virus, 9. Yellow leaf curls virus, 10. Healty.



Gambar 3 Dataset Tomat

2.2.2. Data Preperation

Pada penelitian ini, data preperation sangat penting guna menyesuaikan dataset agar sesuai dengan kebutuhan yang diperlukan pada tahap membangun model. Tahap awal data preparation melibatkan proses pemilihan data untuk melakukan seleksi terhadap informasi yang penting. Hal ini memastikan bahwa dataset yang digunakan dalam penelitian dapat digunakan secara efektif dan sesuai untuk selanjutnya melakukan pembagian data train, validasi, dan test serta membangun model [13].

2.2.3. Pembagian Data Pelatihan, Validasi dan Pengujian

Dalam penelitian ini, terdapat 10.000 gambar dalam dataset pelatihan, 7.000 gambar dalam dataset validasi dan 500 gambar dalam dataset pengujian. Dari 10.000 gambar dalam dataset pelatihan, 1.000 gambar termasuk dalam kategori sehat dan 1.000 gambar termasuk dalam setiap kategori penyakit tomat yang dijelaskan di atas. Pada dataset validasi, setiap kelas memiliki 700 gambar dan dataset pengujian memiliki 50 gambar pada setiap kelasnya [8] [14].

2.2.4. DenseNet201

Pada tahap ini menggunakan algoritma DenseNet201 untuk melakukan identifikasi pada penyakit daun tanaman tomat. Dense Convolutional Network (DenseNet) merupakan salah satu model arsitektur Deep Learning yang menghubungkan setiap layer beserta feature-maps ke seluruh layer berikutnya. Layer berikutnya akan menerima input feature-maps dari seluruh layer sebelumnya. DenseNet menghubungkan setiap lapisan/blok ke setiap lapisan/blok lainnya dengan cara umpan maju. Sedangkan jaringan konvolusional tradisional dengan L lapisan memiliki koneksi L - satu antara setiap lapisan dan lapisan berikutnya jaringan memiliki koneksi langsung L (L + 1)/2. DenseNet memiliki beberapa keunggulan menarik: meringankan masalah gradien-gradien, memperkuat penyebaran fitur, mendorong penggunaan kembali fitur, dan secara substansial mengurangi jumlah parameter [15] [16].

Penelitian ini menggunakan arsitektur DenseNet201 sebagai model dasar dengan menggunakan bobot dari "imagenet" dan menambahkan beberapa lapisan pada lapisan berikutnya. Penambahan lapisan ini termasuk lapisan average pooling, lapisan dropout, lapisan dense, dan fungsi aktivasi. Average pooling menghasilkan nilai rata-rata dari bagian gambar yang tercakup oleh kernel. Penggunaan dropout, yang terbukti dapat meningkatkan akurasi, juga berfungsi sebagai tindakan untuk mencegah overfitting dalam penelitian.

Adapun lapisan – lapisan yang digunakan pada penelitian yaitu seperti:

1. Convolution Layer

Convolution layer adalah komponen inti dalam arsitektur Convolutional Neural Network (CNN) [17]. Layer ini terdiri dari berbagai filter yang diinisialisasi secara acak untuk menjalankan operasi konvolusi, bertujuan untuk mengekstraksi fitur dan memahami representasi gambar input. Pada layer konvolusi, neuron disusun membentuk *feature maps*. Setiap neuron pada feature map sebagai receptive field, terhubung pada neuron-neuron dari convolution layer sebelumnya melalui serangkaian bobot yang dilatih, biasa juga disebut dengan filter bank [18]. Proses *Convolution* menggunakan rumus sebagai berikut [19]:

$$h(x,y)=f(x,y)*g(x,y)$$

2. Pooling

Kemudian dilakukan proses pooling. Jenis yang digunakan pada penelitian ini adalah *average pooling*. Proses ini membagi output menjadi beberapa grid kecil untuk mengambil nilai rata-rata dari setiap grid yang ada pada citra untuk menyusun matriks yang sudah direduksi.

3. ReLu sebagai optimizer

ReLu merupakan fungsi aktivasi yang digunakan pada neural network untuk mengubah nilai x menjadi 0 jika nilai x tersebut bernilai negatif, sedangkan sebaliknya untuk nilai x tetap dipertahankan apabila nilai tidak kurang dari 0 [20] [21]. Proses ReLu menggunakan persamaan seperti berikut:

$$f(xi) = (0, xi)$$

4. Cross entropy sebagai loss function

Cross entropy digunakan untuk mengukur entropi relatif antara dua distribusi probabilitas pada data yang sama, cara kerjanya mengurangi log negatif dari dataset. Proses *Cross entropy* menggunakan rumus seperti berikut:

$$L_{cross-entropy} = - \sum_j y_i \log \log (\hat{y})$$

5. Softmax diterapkan pada lapisan terakhir.

Digunakan untuk menghitung distribusi probabilitas dari vektor bilangan real. Fungsi Softmax menghasilkan output yang merupakan kisaran nilai antara 0 dan 1, dengan jumlah probabilitas sama dengan 1 [22]. Proses *Softmax* menggunakan rumus sebagai berikut:

$$f(xi) = \frac{\exp(xi)}{\sum_j \exp^{xi}}$$

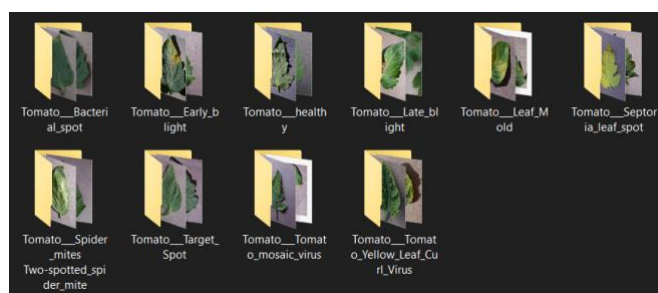
2.3. Evaluasi dan Kesimpulan

Setelah penerapan algoritma DenseNet201 dilakukan dan mendapatkan hasil dari data, kemudian melakukan pengujian dengan mengidentifikasi penyakit daun tomat, setelah itu kinerja model dan hasil pengujian di evaluasi dengan menggunakan confusion matrix. Evaluasi hasil dilihat dari empat aspek yaitu precision, recall, dan accuracy dan F1-score [23]. Confusion matrix digunakan untuk melihat keakurasian atau performa yang dihasilkan dari algoritma yang digunakan, sehingga dapat menarik kesimpulan dari hasil implementasi algoritma dan dapat membandingkan dengan arsitektur VGG16, InceptionV3, dan MobileNet pada penelitian sebelumnya.

3. HASIL DAN PEMBAHASAN

3.1. Persiapan Dataset

Pada penelitian ini dibutuhkan inputan data sebagai pemodelan, Data yang akan digunakan pada penelitian ini adalah data gambar citra daun tanaman tomat yang diambil dari website https://www.tensorflow.org/datasets/catalog/plant_village. Data gambar citra daun tomat yang akan digunakan ini berjumlah 18.835 gambar yang terdiri dari 10 kelas yaitu Tomato Bacterial spot, Tomato Early blight, Tomato healthy, Tomato Late blight, Tomato Leaf Mold, Tomato Septoria leaf spot, Tomato Spider mites, Tomato Target Spot, Tomato mosaic virus, Tomato Yellow Leaf Curl Virus. Data yang didapat masih data mentah yang belum dibagi menjadi data train, validation, dan test seperti pada Gambar 4.



Gambar 4 Dataset Tomat

4.2. Data Preperation

Setelah pengumpulan data maka dilakukan pembagian data untuk train, validation, test sebanyak 10.000 gambar dalam dataset train, 7.000 gambar dalam dataset validation dan 500 gambar dalam dataset test. Dari 10.000 gambar dalam dataset train, 1.000 gambar termasuk dalam kategori sehat dan 1.000 gambar Universitas Muhammadiyah Riau 23 termasuk dalam setiap kategori penyakit tomat. Pada dataset validasi, setiap kelas memiliki 700 gambar dan dataset pengujian memiliki 50 gambar pada setiap kelasnya. Hasil pembagian keseluruhan pada dataset disusun menjadi bentuk tabel.

Tabel 1 Pembagian Dataset

No	Kelas	Train	Validation	Test
1	Tomato_Bacterial_spot	1.000	700	50
2	Tomato_Early_blight	1.000	700	50
3	Tomato_Late_blight	1.000	700	50
4	Tomato_Leaf_Mold	1.000	700	50
5	Tomato_Septoria_leaf_spot	1.000	700	50
6	Tomato_Spider_mitesTwo-spotted_spider_mite	1.000	700	50
7	Tomato_Target_Spot	1.000	700	50
8	Tomato_Tomato_Yellow_Leaf_Curl_Virus	1.000	700	50
9	Tomato_Tomato_mosaic_virus	1.000	700	50
10	Tomato_healthy	1.000	700	50
	Total	10.000	7.000	500

3.3. Membangun Model DenseNet201

Setelah kita melakukan pembagian data train, validation, dan test selanjutnya yaitu melakukan klasifikasi menggunakan google colabs. Pada tahap melakukan klasifikasi algortima yang digunakan adalah DenseNet201.

Selanjutnya dilakukan membangun mengkonfigurasi, dan mengkompilasi model jaringan saraf menggunakan arsitektur DenseNet201 sebagai basisnya, dengan beberapa lapisan tambahan di bagian atasnya untuk klasifikasi data. Arsitektur ini kemudian dikompilasi dengan konfigurasi optimizer, fungsi loss, dan metrik evaluasi yang sesuai.

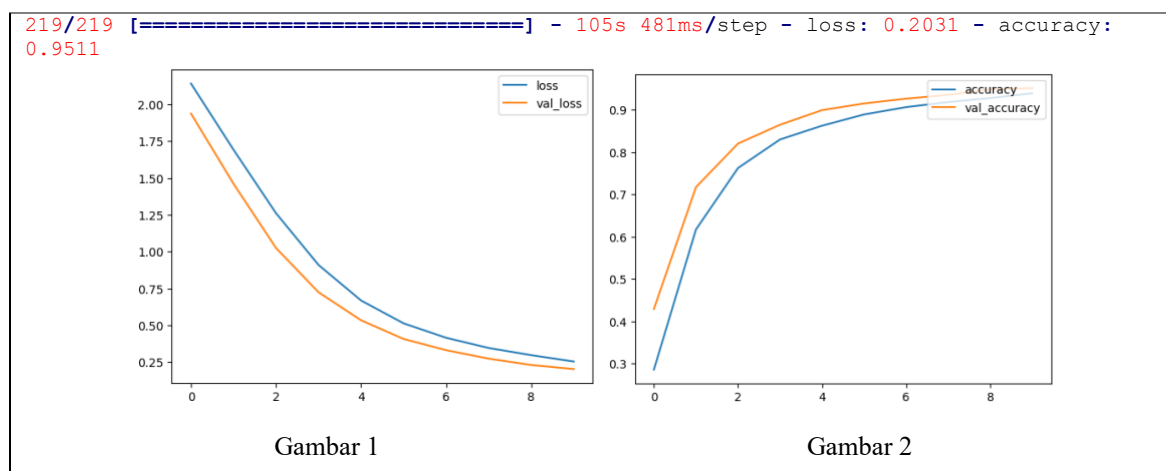
Kemudian dilakukan melatih model jaringan saraf menggunakan data yang diberikan oleh generator pelatihan (train_generator) dan untuk mengevaluasi performa model menggunakan data validasi yang diberikan oleh generator validasi (validation_generator).

```
Epoch 10/10
313/313 [=====] - 291s 927ms/step - loss: 0.2547 - accuracy:
0.9389 - val loss: 0.2036 - val accuracy: 0.9509
```

Pada hasil pelatihan model ini log pelatihan dari sebuah model DenseNet201 selama 10 epoch (siklus pelatihan). Setiap epoch menampilkan beberapa metrik kinerja, termasuk loss (kerugian) dan accuracy (akurasi), baik untuk data pelatihan maupun data validasi (val_loss dan val_accuracy). Misalnya, pada Epoch 1/10, model memiliki loss sebesar 2.1416 dan akurasi sebesar 0.2858 pada data pelatihan, sementara pada data validasi, val_loss sebesar 1.9381 dan val_accuracy sebesar 0.4291. Proses ini berlanjut hingga Epoch 10/10, di mana terlihat bahwa model telah meningkat dalam kinerja, dengan loss yang lebih rendah dan akurasi yang lebih tinggi pada kedua data pelatihan dan data validasi. Hal ini menunjukkan bahwa model terus belajar dari data dan meningkat seiring dengan berjalannya epoch

3.4 Evaluasi Model

Setelah dilakukan pelatihan data selanjutnya dilakukan evaluasi pada hail model yang telah dilatih dengan memvisualisasikan bagaimana loss dan akurasi model berubah selama proses peltihan, serta memantau kinerja model pada data validasi.



Berdasarkan plot grafik yang diatas, ini menunjukkan bahwa gambar 1 (Loss) menunjukkan plot loss atau error fungsi selama proses pelatihan model. Garis biru merepresentasikan loss pada data pelatihan yang menurun dari sekitar 1,75 di awal hingga 0,25 di akhir pelatihan. Garis oranye merupakan loss pada data validasi yang juga menurun dari 1,5 menjadi 0,4. Penurunan loss yang stabil pada kedua garis ini mengindikasikan model dapat belajar dengan baik dari data pelatihan.

Kemudian gambar 2 menampilkan plot akurasi model selama pelatihan. Garis biru adalah akurasi pada data pelatihan yang meningkat dari 0,4 atau 40% di awal hingga 0,95 atau 95% di akhir pelatihan. Sedangkan garis oranye merupakan akurasi pada data validasi yang naik dari 0,4 menjadi 0,9 atau 90%. Peningkatan akurasi yang konsisten pada kedua garis menunjukkan model memperoleh performa klasifikasi yang semakin baik seiring berjalannya proses pelatihan

3.5 Analisis Hasil

Selanjutnya dilakukan evaluasi performa model klasifikasi menggunakan berbagai metrik evaluasi seperti confusion matrix, akurasi, presisi, recall, dan F1-score, ini dilakukan untuk memahami seberapa baik model klasifikasi bekerja dan untuk mengidentifikasi area di mana model dapat ditingkatkan.

Tabel 2 Hasil Evaluasi

Kelas	Test %		
	F1-score	Precision	Recall
Tomato_Bacterial_spot	0.94	0.94	0.94
Tomato_Early_blight	0.92	0.96	0.88
Tomato_Late_blight	0.89	0.90	0.88
Tomato_Leaf_Mold	0.96	0.96	0.96
Tomato_Septoria_leaf_spot	0.96	0.93	1.00
Tomato_Spider_mitesTwo spotted_spider_mite	0.95	0.96	0.94
Tomato_Target_Spot	0.94	0.91	0.98
Tomato_Tomato_Yellow_Leaf_Curl_Virus	0.98	1.00	0.96
Tomato_Tomato_mosaic_virus	1.00	1.00	1.00
Tomato_healthy	0.98	0.98	0.98
Overall Precision	0.95		
Overall Recall	0.95		
Overall F1-score	0.95		
Accuracy	95.20%		

Pada bagian tabel 2 ini menunjukkan hasil evaluasi performa model dalam mengklasifikasikan 10 kelas penyakit pada daun tomat yang setiap baris mewakili satu kelas, Misalnya, untuk kelas "Tomato_Bacterial_spot", model memiliki F1-score 0.94, precision 0.94, dan recall 0.94, yang mengindikasikan performa yang sangat baik dalam mengklasifikasikan kelas ini. Pada bagian bawah tabel menunjukkan metrik keseluruhan, yaitu Overall Precision 0.95, Overall Recall 0.95, Overall F1-score 0.95, dan Akurasi 95.20%.

Selanjutnya, membandingkan hasil penelitian ini pada penelitian sebelumnya, apakah arsitektur DenseNet201 ini memiliki pengaruh pada performa klasifikasi model VGG16, InceptionV3, dan MobileNet. Untuk hasil perbandingan sebagaimana terdapat pada tabel 3 berikut ini.

Tabel 3 Perbandingan Evaluasi Dengan Penelitian Sebelumnya [8]

Arsitektur	Akurasi (%)
VGG16	77.2%
InceptionV3	63.4%
MobileNet	63.75%
CNN Modification	91,2%
DenseNet201	95.20%

Tabel 3 ini membandingkan akurasi dari model yang digunakan dalam penelitian ini (DenseNet201) dengan arsitektur model lain yang digunakan dalam penelitian sebelumnya, yaitu VGG16, InceptionV3, MobileNet dan model usulan dari penelitian tersebut. Dari tabel 4.2, kita dapat melihat bahwa model DenseNet201 yang digunakan dalam penelitian ini memiliki akurasi tertinggi yaitu 95.20%, jauh lebih tinggi dibandingkan VGG16 (77.2%), InceptionV3 (63.4%), MobileNet (63.75%), dan CNN Modification (91,2%). Hal ini karena arsitekturnya yang inovatif menggunakan konsep dense connections. Setiap lapisan dalam jaringan terhubung langsung ke setiap lapisan lain, memungkinkan transfer fitur yang lebih baik serta memperkuat propagasi gradien selama pelatihan. Selain itu, DenseNet201 memiliki regularisasi intrinsik yang mengurangi overfitting, efisiensi parameter yang lebih tinggi meskipun memiliki lebih banyak lapisan, serta pelatihan yang lebih efisien.

4. KESIMPULAN

Berdasarkan hasil penelitian Identifikasi Penyakit Daun Tanaman Tomat Melalui Citra Daun Menggunakan DenseNet201 dapat disimpulkan bahwa Model DenseNet201 yang dibangun mencapai akurasi 95,20% pada data pengujian, dengan nilai F1-Score keseluruhan 0,95. Dibandingkan dengan penelitian sebelumnya yang menggunakan arsitektur VGG16 akurasi sebesar 77,2%, InceptionV3 akurasi sebesar 63,4%, MobileNet akurasi sebesar 63,75%, dan CNN Modification 91,2%, model DenseNet201 menunjukkan peningkatan performa yang signifikan dalam mengklasifikasikan penyakit tanaman tomat..

DAFTAR PUSTAKA

- [1] V. K. Vishnoi, K. Kumar, and B. Kumar, *Plant disease detection using computational intelligence and image processing*, vol. 128, no. 1. Springer Berlin Heidelberg, 2021. doi: 10.1007/s41348-020-00368-0.
- [2] M. Faried, G. El Mirzaq, and R. Helilintar, "Sistem Pakar Diagnosa Penyakit Tanaman Tomat Menggunakan Metode Certainty Factor," *Pros. SEMNAS INOTEK (Seminar Nas. Inov. Teknol.*, vol. 5, no. 2, pp. 230–235, Aug. 2021, doi: 10.29407/INOTEK.V5I2.1134.
- [3] T. Ha, "Produktivitas Tomat Menurut Provinsi , 2012-2016," vol. 2016, p. 2016, 2016.
- [4] N. I. Yahya, S. Lestanti, and S. N. Budiman, "SISTEM PAKAR DIAGNOSIS HAMA DAN PENYAKIT TANAMAN AGLAONEMA," vol. 6, no. 2, pp. 734–741, 2022.
- [5] D. Ali and S. Frimpong, "Artificial intelligence, machine learning and process automation: existing knowledge frontier and way forward for mining sector," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 6025–6042, 2020, doi: 10.1007/s10462-020-09841-6.
- [6] G. Abdurrahman, "Klasifikasi Penyakit Diabetes Melitus Menggunakan Adaboost Classifier," *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.*, vol. 7, no. 1, pp. 59–66, 2022, doi: 10.32528/justindo.v7i1.4949.
- [7] P. Pradhan, B. Kumar, and S. Mohan, "Comparison of various deep convolutional neural network models to discriminate apple leaf diseases using transfer learning," *J. Plant Dis. Prot.*, vol. 129, no. 6, pp. 1461–1473, 2022, doi: 10.1007/s41348-022-00660-1.
- [8] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 293–301, 2020, doi: 10.1016/j.procs.2020.03.225.
- [9] S. D. Meena, M. Susank, T. Guttula, S. H. Chandana, and J. Sheela, "Crop Yield Improvement with Weeds, Pest and Disease Detection," *Procedia Comput. Sci.*, vol. 218, no. 2022, pp. 2369–2382, 2022, doi: 10.1016/j.procs.2023.01.212.
- [10] P. Sitompul, H. Okprana, A. Prasetyo, and G. Artikel, "Identifikasi Penyakit Tanaman Padi Melalui Citra Daun Menggunakan DenseNet 201 Identification of Rice Plant Diseases Through Leaf Image Using DenseNet 201 Article Info ABSTRAK," *JOMLAI J. Mach. Learn. Artif. Intell.*, vol. 1, no. 2, pp. 143–150, 2022, doi: 10.55123/jomlai.v1i2.889.
- [11] E. Retnoningsih and R. Pramudita, "Mengenal Machine Learning Dengan Teknik Supervised dan Unsupervised Learning Menggunakan Python," vol. 7, no. 2, pp. 156–165, 2020.
- [12] A. Fatih, M. Najamuddin, and D. Miharja, "Implementasi Sistem Pengenalan Candi Kecil Di Yogyakarta Menggunakan Machine Learning Berbasis Cloud," *Pros. Sains dan Teknol.*, vol. 1, no. 1, pp. 96–102, Aug. 2022, Accessed: Dec. 21, 2023. [Online]. Available: <https://www.jurnal.pelitabangsa.ac.id/index.php/SAINTEK/article/view/1155>
- [13] D. P. Utomo and M. Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung," *J. Media Inform. Budidarma*, vol. 4, no. 2, p. 437, 2020, doi: 10.30865/mib.v4i2.2080.
- [14] R. S. Tantika and A. Kudus, "Penggunaan Metode Support Vector Machine Klasifikasi Multiclass pada Data Pasien Penyakit Tiroid," *Bandung Conf. Ser. Stat.*, vol. 2, no. 2, pp. 159–166, 2022, doi: 10.29313/bcss.v2i2.3590.
- [15] J. Pardede and D. A. L. Putra, "Implementasi DenseNet Untuk Mengidentifikasi Kanker Kulit Melanoma," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, pp. 425–433, 2020, doi: 10.28932/jutisi.v6i3.2814.
- [16] I. N. Husna *et al.*, "Rancang Bangun Sistem Deteksi Dan Perhitungan Jumlah Orang Menggunakan Metode Convolutional Neural Network (CNN)," pp. 3–8.
- [17] Luqman Hakim, Z. Sari, and Handhajani, "Klasifikasi Citra Pigmen Kanker Kulit Menggunakan Convolutional Neural Network," vol. 1, no. 10, pp. 379–385, 2021.
- [18] P. A. Nugroho, I. Fenriana, R. Arijanto, and M. Kom, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA EKSPRESI MANUSIA," vol. 1, 2020.
- [19] N. Khunafa Qudsi, R. A. Asmara, and A. R. Syulistyo, "Identifikasi Citra Tulisan Tangan Digital Menggunakan Convolutional Neural Network (CNN)," *Semin. Inform. Apl.*, pp. 48–53, 2019.
- [20] A. Faizin, A. Tri Arsanto, Moch. Lutfi, and A. Rochim Musa, "Deep Pre-Trained Model Menggunakan Arsitektur Densenet Untuk Identifikasi Penyakit Daun Padi," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 6, no. 2, pp. 615–621, 2022, doi: 10.36040/jati.v6i2.5475.
- [21] A. W. Salehi *et al.*, "A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope," *Sustain.*, vol. 15, no. 7, 2023, doi: 10.3390/su15075930.
- [22] D. Sumarlie *et al.*, "Pengenalan Kue Tradisional Indonesia," vol. 2, pp. 164–171, 2022.
- [23] P. R. Sihombing and I. F. Yulianti, "Penerapan Metode Machine Learning dalam Klasifikasi Risiko Kejadian Berat Badan Lahir Rendah di Indonesia," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 20, no. 2, pp. 417–426, 2021, doi: 10.30812/matrik.v20i2.1174.