

Klasifikasi Citra Penyakit Tanaman Pada Daun Paprika Dengan Metode *Transfer Learning* Menggunakan DenseNet-201**Vilvilia Salim¹, Asrul Abdullah^{2*}, Putri Yuli Utami³**

vilvilia.salim@unmuhpnk.ac.id, asrul.abdullah@unmuhpnk.ac.id, putriyuli@unmuhpnk.ac.id

Universitas Muhammadiyah Pontianak

Informasi Artikel

Diterima : 3 Feb 2024

Direview : 1 Apr 2024

Disetujui : 23 Apr 2024

Abstrak

Penyakit bercak daun yang disebabkan oleh bakteri *Xanthomonas campestris* pv. *vesicatoria* merupakan salah satu penyakit penting pada tanaman paprika di Indonesia. Penyakit ini dapat menurunkan kualitas dan kuantitas hasil panen paprika. Metode yang digunakan yaitu *transfer learning* dengan menggunakan model DenseNet-201. Penelitian ini menggunakan data gambar daun paprika yang terinfeksi dan tidak terinfeksi sebanyak 4.876 gambar. Data tersebut dibagi menjadi data latih, data validasi, dan data uji. Hasil penelitian menunjukkan bahwa model transfer learning mampu mendeteksi penyakit bercak daun pada paprika dengan akurasi keseluruhan sekitar 99.5%. Evaluasi model terhadap kelas "Bacterial Spot" dan "Healthy" menghasilkan *precision*, *recall*, dan *F1-score* rata-rata sekitar 99.5%. Penelitian ini menunjukkan bahwa metode *transfer learning* dapat digunakan sebagai sistem deteksi penyakit tanaman yang efektif dan efisien.

Kata KunciCnn, DenseNet-201,
Paprika, Penyakit Bercak
Daun, Transfer Learning**Keywords**Cnn, DenseNet-201, Leaf
Spot Disease, Pepper,
Transfer Learning**Abstrak**

Leaf spot disease caused by the bacterium Xanthomonas campestris pv. *vesicatoria* *is one of the significant diseases affecting pepper plants in Indonesia. This disease can reduce the quality and quantity of pepper harvests. The method used in this study is transfer learning using the DenseNet-201 model. The research utilizes a dataset of 4,876 images of infected and uninfected pepper leaves. The data is divided into training, validation, and test sets. The results show that the transfer learning model can detect leaf spot disease in peppers with an overall accuracy of around 99.5%. The evaluation of the model for the classes "Bacterial Spot" and "Healthy" yields an average precision, recall, and F1-score of approximately 99.5%. This research demonstrates that transfer learning can be employed as an effective and efficient plant disease detection system.*

A. Pendahuluan

Penyakit daun paprika dapat disebabkan oleh berbagai faktor, termasuk bakteri, virus, jamur, dan serangga [1]. Salah satu penyakit yang umum terjadi pada paprika adalah bacterial spot, yang disebabkan oleh bakteri *Xanthomonas campestris* [2]. Bakteri *Xanthomonas campestris* pv. *Vesicatoria* adalah patogen tanaman yang menyebabkan penyakit bercak daun pada tanaman paprika. Penyakit ini dapat menyebabkan kerugian yang besar karena dapat menyebar dengan cepat dan menyebabkan kerusakan pada daun, batang, dan buah paprika [3].

Untuk mengklasifikasi penyakit pada daun paprika digunakan salah satu metode *deep-learning* yang merupakan salah satu cabang dari *machine learning* yang menggunakan *neural network* dengan banyak *layer* untuk mempelajari pola dari data yang kompleks, seperti gambar [4]. Sementara itu *transfer learning* adalah teknik *deep learning* yang memanfaatkan pengetahuan yang telah dipelajari dari tugas sebelumnya untuk tugas baru. Dalam *transfer learning*, model *deep learning* yang telah dilatih pada *dataset* besar dapat digunakan sebagai awal yang baik untuk mempelajari *dataset* baru yang lebih kecil. Dengan memanfaatkan pengetahuan yang telah dipelajari dari tugas sebelumnya, *transfer learning* dapat menghasilkan model *deep learning* yang lebih baik [5].

Deep learning, khususnya Convolutional Neural Network (CNN), mampu memberikan hasil signifikan dalam mengenali objek gambar, seperti dalam masalah klasifikasi. Beberapa penelitian menggunakan metode *transfer learning* untuk mengklasifikasikan penyakit pada tanaman lain seperti kentang [6], kopi [7], padi [4], dan apel [8]. Dari hasil dari penelitian tersebut menunjukkan bahwa metode *transfer learning* memiliki tingkat akurasi yang tinggi dalam mengklasifikasikan penyakit pada tanaman.

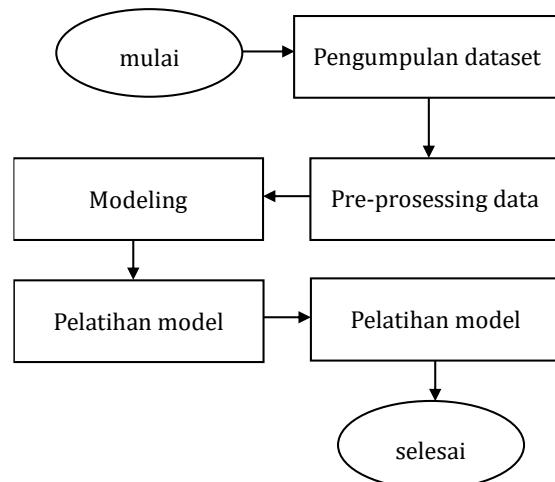
Pada penelitian ini, klasifikasi menggunakan metode *transfer learning* dengan model DenseNet201. Model DenseNet201 dipilih karena memiliki keunggulan dalam klasifikasi citra. Sebuah penelitian menunjukkan bahwa DenseNet201 telah terbukti memberikan hasil yang sangat baik dalam klasifikasi citra dibanding model yang lain seperti AlexNet, LeNet, VGGNET, Resnet dan GoogleNet [9].

Pada penelitian sebelumnya [10], digunakan *dataset* yang dibagi menjadi daun sehat dan daun yang terinfeksi bercak bakteri. Implementasi CNN dengan *transfer learning* menggunakan arsitektur VGG16 menghasilkan hasil akurasi tes sebesar 90%, akurasi pelatihan 97% dengan loss 8,5%, akurasi validasi sebesar 97,5% dengan loss 6,9%.

Berdasarkan uraian diatas, peneliti melakukan pemodelan untuk mengklasifikasi penyakit tanaman pada daun paprika dengan menggunakan metode *transfer learning* dengan model DenseNet201 dan melakukan pengujian perbandingan terhadap *optimizer*, *learning rate*, *epoch* dan *batch size* yang diharapkan dapat memiliki nilai akurasi yang lebih tinggi.

B. Metode Penelitian

Metode penelitian ini diilustrasikan melalui sebuah diagram alir yang menjelaskan langkah-langkah utama yang dapat dilihat pada Gambar 1.

**Gambar 2.** Diagram alir

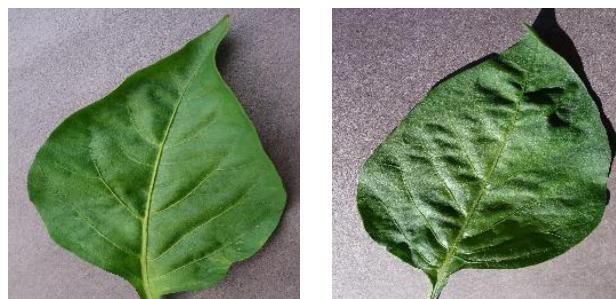
1. Pengumpulan Dataset

Data yang digunakan dalam penelitian ini adalah *dataset* citra penyakit tanaman pada daun paprika terdiri dari 2 kelas yaitu Sehat (*Healthy*) dan Bercak bakteri (*bacterial spot*). *Dataset* yang digunakan berasal dari *dataset* publik menggunakan *dataset* yang sudah tersedia dari platform Kaggle yang disebut “*New Plant Diseases Dataset*”. *Dataset* tersebut memiliki berbagai jenis tanaman seperti jangung ,padi, tomat, kentang, dan sebagainya. Pada penelitian ini digunakan label tanaman paprika yang berjumlah 4.876 citra. Jumlah data yang digunakan untuk data latih berjumlah 3.701 dan data validasi berjumlah 975 dan data uji berjumlah 200. Berikut jumlah proporsi tiap data didalam dataset pada Tabel 1.

Tabel 1. Jumlah dataset

| No | Kelas | Latih | Validasi | Uji |
|----|----------------|-------|----------|-----|
| 1 | Healthy | 1888 | 497 | 100 |
| 2 | bacterial spot | 1813 | 478 | 100 |

Berikut ini adalah contoh citra yang digunakan Gambar 2 dan Gambar 3 :

**Gambar 2.** Contoh Daun Sehat



Gambar 3. Contoh Daun Bercak Bakteri

2. Data preprocessing

Prapemrosesan data adalah tahap awal dalam pengolahan data yang bertujuan untuk mempersiapkan data mentah agar dapat diolah lebih lanjut. Tahap ini meliputi beberapa proses, diantaranya, *load data* atau *input dataset* menjadi 3 bagian, konfigurasikan *dataset* untuk kinerja, gunakan augmentasi data, dan normalisasi [10].

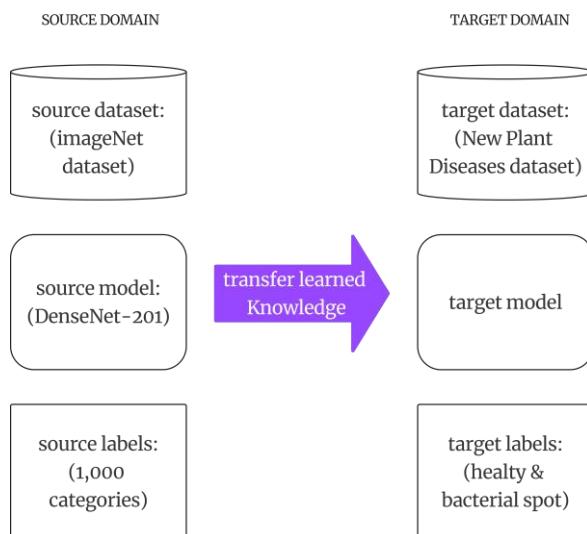
Pre-processing pada pengklasifikasian daun paprika menggunakan CNN bertujuan untuk memperbaiki kualitas citra dan mempersiapkan citra agar dapat diolah oleh CNN dengan baik [11]. Pada tahap pre-processing, dilakukan *resize* untuk mengubah ukuran citra agar memiliki ukuran yang sama dan sesuai dengan ukuran *input* CNN dengan ukuran 224x224 [12]. Selain itu, dilakukan normalisasi untuk memotong citra daun paprika menjadi ukuran yang sama dan memperbaiki kontras citra dengan cara membagi nilai intensitas tiap piksel dengan 255 sehingga rentang nilai tiap piksel adalah -1 sampai 1 [13]. Secara matematis normalisasi dihitung dengan persamaan:

$$\text{Normalization} = \frac{x - \mu}{\sigma} \quad (1)$$

Di mana x adalah *vektor* fitur asli, μ adalah *mean*, dan σ adalah standar deviasi.

3. Modeling

Modeling dilakukan menggunakan pendekatan *transfer learning*, dalam tahap modeling dilakukan *import pre-trained* model yang digunakan untuk menjadi model baru dalam melakukan tugas baru dengan *dataset* baru [9]. Dalam penelitian ini, model *pre-trained* yang digunakan adalah DenseNet-201, yang merupakan model arsitektur Convolutional Neural Network (CNN) yang telah dilatih sebelumnya pada *dataset imangenet*. *Transfer learning* dapat digunakan untuk mengklasifikasikan citra daun paprika pada domain target dengan menerapkan hasil dari lapisan ekstraksi fitur pada *domain* sumber ke lapisan ekstraksi fitur pada *domain* target[14]. Berikut ilustrasi dasar *transfer learning* pada Gambar 4.

**Gambar 4.** Ilustrasi *transfer learning*

Lapisan terakhir dari model yang sebelumnya telah dilatih harus diganti dengan lapisan klasifikasi yang baru. Hal ini memungkinkan model fokus pada pelatihan lapisan klasifikasi baru [13]. Penggantian dengan lapisan baru yang menggunakan fungsi aktivasi sigmoid. Fungsi aktivasi tersebut dapat digunakan untuk mengklasifikasikan dua label, yang akan disesuaikan dengan *output* terkait penyakit pada daun paprika.

4. Training model

Pelatihan dalam penelitian ini dilakukan melalui empat skenario berbeda untuk mengidentifikasi parameter yang menghasilkan kinerja terbaik. Parameter yang dievaluasi melibatkan perbandingan terhadap *optimizer*, *learning rate*, *epoch*, dan *batch Size*. Berikut hasil akhir pemilihan hyperparameter yang digunakan pada Tabel 2 [15].

Tabel 2. *Hyperparameter*

| No | <i>Optimizer</i> | <i>Learning Rate</i> | <i>Epoch</i> | <i>Batch Size</i> |
|----|------------------|----------------------|--------------|-------------------|
| 1 | <i>SGD</i> | 0,001 | 10 | 16 |
| 2 | <i>RMSprop</i> | 0,0001 | 20 | 32 |
| 3 | <i>Adam</i> | 0,00001 | 30 | 64 |
| 4 | <i>Adagrad</i> | 0,000001 | 40 | 128 |

5. Evaluasi

Setelah proses pelatihan model selesai, langkah selanjutnya adalah mengevaluasi model yang telah dilatih. Evaluasi model dilakukan dengan menggunakan *dataset* yang terpisah dari *dataset* yang digunakan untuk pelatihan model. Untuk mengukur kinerja model yang digunakan untuk klasifikasi, digunakan *confusion matrix* (cm). Evaluasi model dilakukan dengan memantau jumlah *true positive*, *true negative*, *false positive*, dan *false negative* [16]. Dari data tersebut dapat dihitung akurasi, *recall*, *presisi* dan *F1-score* [17]. Masing-masing disajikan dari persamaan (2) ke persamaan (5).

$$\text{Accuracy} = \frac{(TN+TP)}{(TN+TP+FN+FP)} \quad (2)$$

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (3)$$

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (4)$$

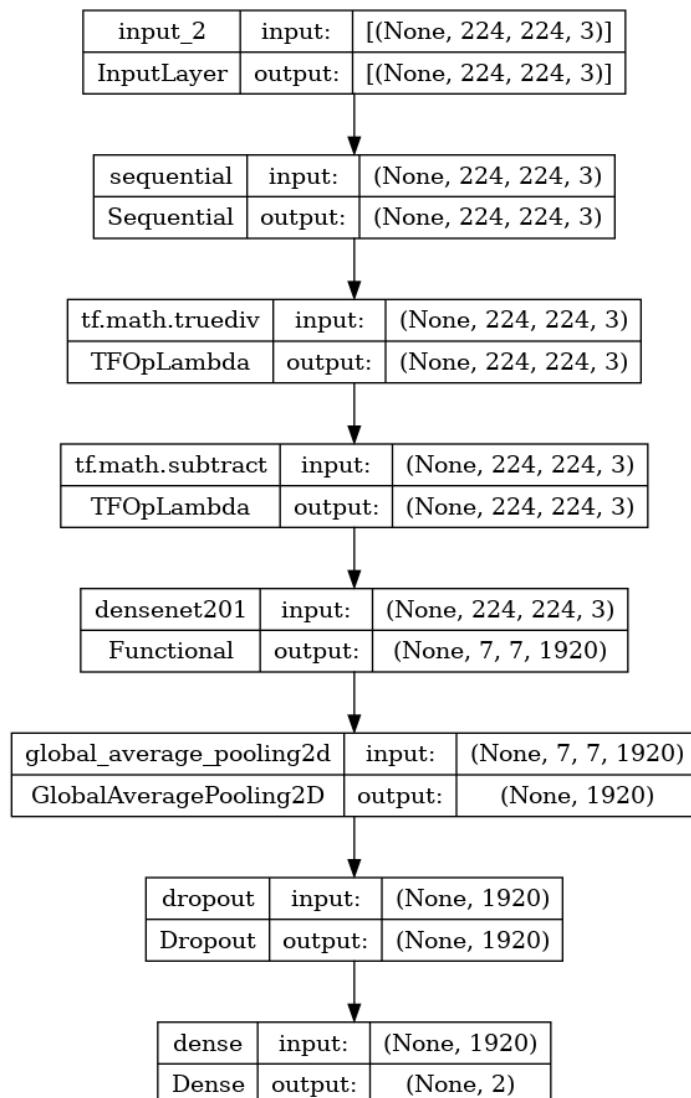
$$F1 - score = 2 \times \frac{\text{Precision} \times \text{Recall}}{(\text{Precision} + \text{Recall})} \quad (5)$$

Istilah-istilah berikut digunakan untuk menjelaskan kinerja model:

- a. *True positive* (TP): data citra daun paprika positif yang diprediksi dengan benar
- b. *True negative* (TN): data citra daun paprika negatif yang diprediksi dengan benar
- c. *False positive* (FP): data citra daun paprika negatif yang salah diprediksi positif
- d. *False negative* (FN): data citra daun paprika positif yang salah diprediksi sebagai negatif

C. Hasil dan Pembahasan

Hasil dan pengujian penelitian tentang pengujian sistem pada klasifikasi citra penyakit daun tanaman paprika dilakukan dengan menggunakan arsitektur DenseNet-201 dan metode *transfer learning*. Tujuan dari pengujian sistem ini adalah untuk memastikan bahwa penggunaan parameter akurasi, recall, presisi, dan F1-score mencapai hasil terbaik dalam klasifikasi citra daun paprika. Berikut hasil grafik representasi visual dari arsitektur model menggunakan model DenseNet-201 pada Gambar 5.

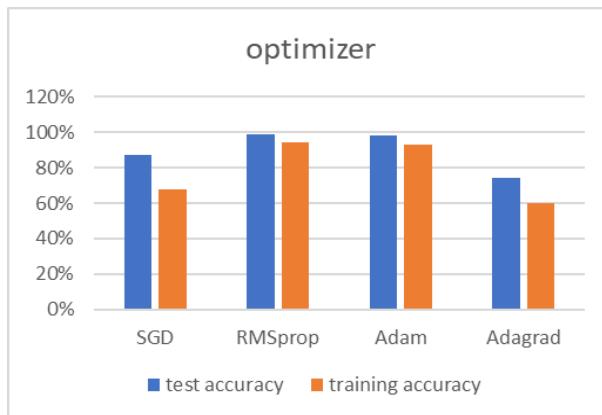
**Gambar 5.** Visual dari arsitektur model

1. Hasil Skenario Pengujian Sistem

Sistem akan diuji dengan menggunakan empat skenario yaitu perbandingan terhadap *optimizer*, perbandingan terhadap *learning rate*, perbandingan terhadap *epoch* dan perbandingan terhadap *batch size*.

1.1 Skenario 1: *Optimizer*

Pemilihan *optimizer* adalah tahap yang sangat penting dalam proses pelatihan model, karena dapat memiliki dampak signifikan terhadap kinerja keseluruhan model. Pada skenario pertama, kami melakukan pengujian terhadap beberapa *optimizer* dengan tujuan untuk menemukan *optimizer* yang paling efisien dalam tugas klasifikasi citra daun paprika. Algoritma optimasi yang diteliti meliputi SGD, ADAM, RMSprop, dan ADGRAD. Hasil pengujian terhadap *optimizer* pada Gambar 6.

**Gambar 6. Optimizer**

Gambar 6 menunjukkan bahwa *optimizer RMSprop* juga dikenal sebagai *Root Mean Square Propagation* adalah metode optimasi yang memungkinkan untuk mempertahankan nilai belajar sesuai dengan rata-rata gradien [15]. Selain itu, *optimizer RMSprop* akan digunakan untuk membandingkan *learning rate*, *epoch*, dan *batch size*.

1.2 Skenario 2: *Learning Rate*

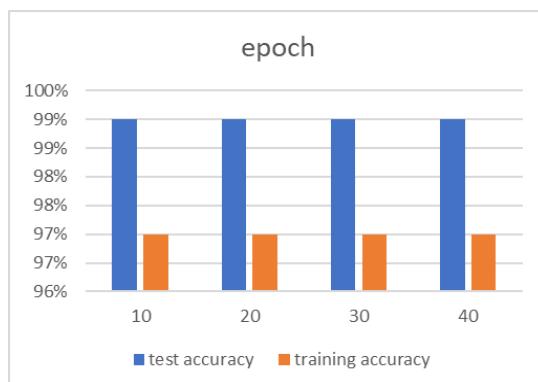
Skenario kedua melibatkan pengujian terhadap berbagai nilai *Learning Rate* menggunakan hasil analisis terbaik dari skenario pertama. Nilai-nilai *Learning Rate* yang diuji dalam skenario ini adalah 0,001, 0,0001, 0,00001, dan 0,000001. Proses pengujian dilakukan dengan memanfaatkan *optimizer RMSprop*, melakukan training selama 10 *epochs*, dan menggunakan *batch size* sebesar 32. Hasil pengujian terhadap learning rate dapat dilihat pada Gambar 7.

**Gambar 7. Learning rate**

Berdasarkan Gambar 7, pengujian terhadap *Learning Rate* 0,001 menunjukkan hasil akurasi yang cukup tinggi, mencapai 97%. Dengan demikian, berdasarkan hasil tersebut, model dapat dinilai sebagai cukup baik karena tingkat akurasinya yang tinggi.

1.3 Skenario 3: *Epoch*

Pengujian terhadap jumlah *epoch* dilakukan dengan menerapkan parameter terbaik yang dihasilkan dari skenario sebelumnya. *epoch* merupakan iterasi berulang dalam proses pelatihan yang bertujuan untuk mencapai nilai akurasi optimal. Jumlah *epoch* yang diuji adalah 10, 20, 30, dan 40, dengan menggunakan *optimizer RMSprop*, nilai *learning rate* 0,001, dan *batch size* 32. Hasil pengujian terhadap jumlah *epochs* dapat dilihat pada Gambar 8.

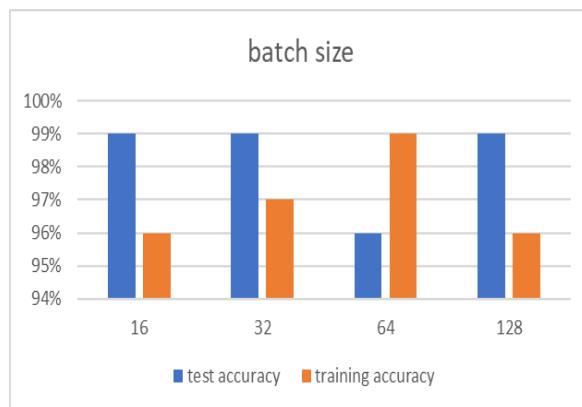


Gambar 8. epoch

Berdasarkan Gambar 8, hasil dari semua *epoch* menunjukkan hasil yang sama rata-rata. Oleh karena itu, peneliti memutuskan untuk memilih *epoch* dengan nilai 10 karena waktu pelatihan lebih efisien dibandingkan dengan nilai *epoch* yang lebih tinggi.

1.4 Skenario 4: Batch Size

Batch Size adalah pembagian *dataset per-batch* pada setiap *epoch* untuk mempercepat proses pelatihan. Skenario terakhir melibatkan pengujian terhadap *batch size* yang paling optimal. *Batch size* yang akan diuji adalah 16, 32, 64, dan 128, dengan menggunakan *optimizer RMSprop*, nilai *learning rate* 0,001, dan jumlah *epochs* sebanyak 10. Hasil pengujian terhadap *batch size* pada Gambar 9.



Gambar 9. Batch Size

Berdasarkan Gambar 9, hasil pengujian menunjukkan bahwa *batch size* yang paling efisien adalah 32. *batch size* 32 menghasilkan nilai *test accuracy* yang tinggi, mencapai 99%, sementara *training accuracy* terbaik dicapai dengan *batch*

size 64, juga mencapai 99%. Namun, peneliti memilih *batch size* 32 karena hasil pengujian menunjukkan kinerja yang baik dengan *batch size* tersebut.

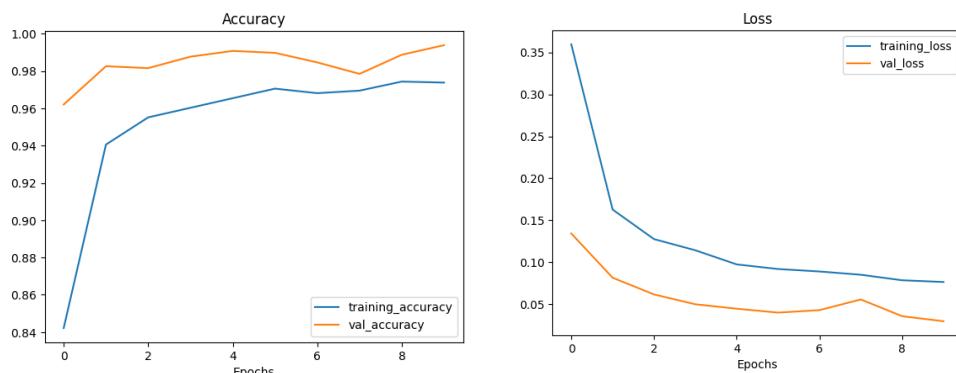
2. Hasil pelatihan

Setelah menguji empat skenario, yaitu *optimizer*, *learning rate*, *epochs*, dan *batch size*, ditemukan bahwa *batch size* memiliki dampak paling signifikan terhadap hasil pengujian. Berikut Pemilihan *Hyperparameter* yang digunakan pada penelitian ini pada Tabel 3.

Tabel 3. Pemilihan *Hyperparameter*

| No | Parameter | Nilai |
|----|----------------------|---------|
| 1 | <i>Optimizer</i> | RMSprop |
| 2 | <i>Learning Rate</i> | 0.001 |
| 3 | <i>Epoch</i> | 10 |
| 4 | <i>Batch Size</i> | 32 |

Berikut hasil pengujian *training* pada penelitian ini sebagai berikut pada Gambar 10 dan Gambar 11:



Gambar 10. Hasil grafik pelatihan

```

Epoch 1/10
116/116 [=====] - 43s 247ms/step - loss: 0.3594 - accuracy: 0.8422 - val_loss: 0.1344 - val_accuracy: 0.9621
Epoch 2/10
116/116 [=====] - 22s 187ms/step - loss: 0.1628 - accuracy: 0.9406 - val_loss: 0.0817 - val_accuracy: 0.9826
Epoch 3/10
116/116 [=====] - 22s 186ms/step - loss: 0.1277 - accuracy: 0.9551 - val_loss: 0.0619 - val_accuracy: 0.9815
Epoch 4/10
116/116 [=====] - 22s 186ms/step - loss: 0.1144 - accuracy: 0.9603 - val_loss: 0.0502 - val_accuracy: 0.9877
Epoch 5/10
116/116 [=====] - 22s 186ms/step - loss: 0.0976 - accuracy: 0.9654 - val_loss: 0.0449 - val_accuracy: 0.9908
Epoch 6/10
116/116 [=====] - 22s 187ms/step - loss: 0.0921 - accuracy: 0.9705 - val_loss: 0.0402 - val_accuracy: 0.9897
Epoch 7/10
116/116 [=====] - 22s 186ms/step - loss: 0.0892 - accuracy: 0.9681 - val_loss: 0.0432 - val_accuracy: 0.9846
Epoch 8/10
116/116 [=====] - 22s 186ms/step - loss: 0.0853 - accuracy: 0.9695 - val_loss: 0.0559 - val_accuracy: 0.9785
Epoch 9/10
116/116 [=====] - 22s 188ms/step - loss: 0.0787 - accuracy: 0.9743 - val_loss: 0.0360 - val_accuracy: 0.9887
Epoch 10/10
116/116 [=====] - 22s 187ms/step - loss: 0.0767 - accuracy: 0.9738 - val_loss: 0.0300 - val_accuracy: 0.9938

```

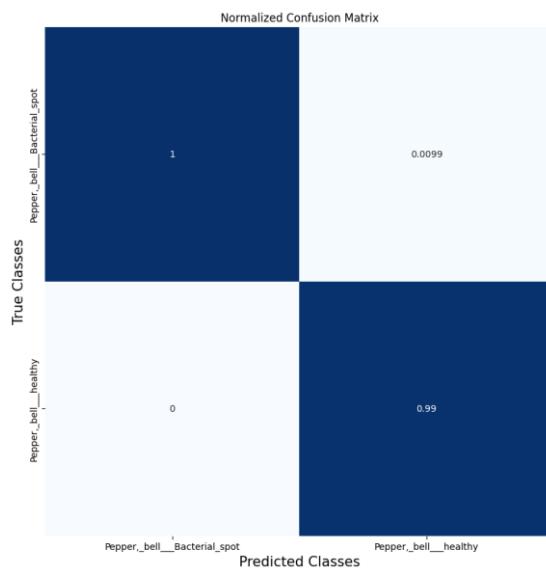
Gambar 11. Hasil pelatihan dengan 10 Epoch

Secara keseluruhan, hasil latihan dengan 10 *epoch* yang diberikan menunjukkan bahwa model tersebut berperforma dengan baik. Dalam hasil *training* ini, *loss* pelatihan menurun dari 0.3594 menjadi 0.0767, menunjukkan bahwa model belajar dan meningkatkan performanya. Akurasi pelatihan meningkat dari 0.8422 menjadi 0.9738, yang menunjukkan bahwa model belajar dengan baik dari data. Penurunan *validation loss* dan peningkatan *validation*

accuracy (dari 0.1344 menjadi 0.0300 dan dari 0.9621 menjadi 0.9938) menunjukkan bahwa model tidak hanya menghafal data pelatihan tetapi juga mampu menggeneralisasi dengan baik pada data baru yang tidak terlihat selama pelatihan.

3. Hasil pengujian

Nilai kinerja sistem (*Recall*, *Precision*, dan *F1-score* menggunakan *Confusion Matrix*) adalah parameter tambahan yang dapat digunakan untuk menentukan kinerja model terbaik. Hasil nilai *confusion matrix* pada Gambar 12.



Gambar 12. Hasil confusion matrix

1. *Actual: Bacterial spot* dan *Predicted: Bacterial spot* (100%): Model berhasil memprediksi 100% sebagai *Bacterial spot* yang sebenarnya merupakan *Bacterial spot* (*True Positive*).
2. *Actual: Bacterial spot* dan *Predicted: Healthy* (0): Model tidak membuat kesalahan dengan memprediksi *instance* yang sebenarnya positif sebagai negatif. Ini adalah hasil yang sangat baik dan menunjukkan bahwa model dapat mendeteksi dengan baik *instance positif* (*False Negative*).
3. *Actual: Healthy* dan *Predicted: Bacterial spot* (0): model membuat kesalahan dengan memprediksi sekitar 0.99% dari *instance* yang sebenarnya negatif sebagai positif (*False Positive*).
4. *Actual: Healthy* dan *Predicted: Healthy* (100): model berhasil memprediksi sekitar 99% dari *instance* yang sebenarnya negatif dengan benar. Ini adalah hasil yang sangat baik dan menunjukkan kemampuan model untuk mengidentifikasi dengan tepat *instance* yang sebenarnya negatif (*True Negative*).

Performa model dapat dilihat dari matriks evaluasi yang mencakup *precision*, *recall*, dan *F1-score*. Berikut adalah Matrix evaluasi untuk hasil yang diberikan pada Gambar 13.

| | precision | recall | f1-score | support |
|------------------------------|-----------|--------|----------|---------|
| Pepper,_bell__Bacterial_spot | 1.00 | 0.99 | 0.99 | 100 |
| Pepper,_bell__healthy | 0.99 | 1.00 | 1.00 | 100 |
| accuracy | | | 0.99 | 200 |
| macro avg | 1.00 | 0.99 | 0.99 | 200 |
| weighted avg | 1.00 | 0.99 | 0.99 | 200 |

Gambar 13. Matrix evaluasi

Berikut rata-rata matriks evaluasi yang mencakup *precision*, *recall*, dan *F1-score* pada Tabel 4 bawah berikut:

Tabel 4. Hasil Rata-Rata Bobot Matrik Evaluasi

| No | Nama | Precision | Recall | F1-Score |
|----|-----------------------|-----------|--------|----------|
| 1 | <i>Bacterial Spot</i> | 1.00 | 0.99 | 0.99 |
| 2 | <i>healthy</i> | 0.99 | 1.00 | 1.00 |
| 3 | Rata rata bobot | 0.995 | 0.995 | 0.995 |

Pada penelitian ini Akurasi sebesar 0.99 menunjukkan seberapa baik model dapat mengklasifikasikan secara keseluruhan dan Menghitung rata-rata dari metrik-metrik untuk setiap kelas, memberikan bobot berdasarkan jumlah *instance* dalam setiap kelas. Weighted average dari 99.5%, menunjukkan performa keseluruhan yang baik, dengan mempertimbangkan distribusi *instance* dalam setiap kelas.

D. Simpulan

Berdasarkan penelitian ini, dapat disimpulkan bahwa penyakit bercak daun dengan metode *transfer learning* dengan Model klasifikasi yang dilatih menunjukkan performa yang sangat baik dengan akurasi keseluruhan sekitar 99%. Evaluasi model terhadap kelas "*Bacterial spot*" dan "*Healthy*" menghasilkan *precision*, *recall*, dan *F1-score* rata-rata sekitar 99.5%, menunjukkan kemampuan model untuk mengidentifikasi dengan sangat baik *instance* di kedua kelas tersebut. Rata-rata bobot untuk metrik evaluasi menunjukkan konsistensi performa model, memberikan keyakinan bahwa model ini dapat diandalkan dan efektif untuk tugas klasifikasi yang diberikan.

E. Referensi

- [1] N. Inaya, S. Meriem, and M. Masriany, "Identifikasi morfologi penyakit tanaman cabai (*Capsicum sp.*) yang disebabkan oleh patogen dan serangan hama lingkup kampus UIN Alauddin Makassar," *Filogeni J. Mhs. Biol.*, vol. 2, no. 1, pp. 8–14, 2022, doi: 10.24252/filogeni.v2i1.27092.
- [2] J. Carroll, "No Title Peppery Leaf Spot: How To Treat Bacterial Leaf Spot On Peppers," 2021. <https://www.gardeningknowhow.com/edible/vegetables/pepper/bacterial-leaf-spot-on-peppers.htm>.
- [3] T. Tâm, N. C. Ú U. Và, C. Ê N. Giao, C. Ngh, and Å N B Ù I Chu, "Uji Daya Hambat *Streptomyces Roseoflavus* Al2 Terhadap *Xanthomonas* Sp. Penyebab

- Penyakit Hawar Daun Bakteri (Hdb) Pada Tanaman Padi (*Oryza sativa L.*)," vol. 01, pp. 1–23, 2016.
- [4] S. Yuliany, Aradea, and Andi Nur Rachman, "Implementasi Deep Learning pada Sistem Klasifikasi Hama Tanaman Padi Menggunakan Metode Convolutional Neural Network (CNN)," *J. Buana Inform.*, vol. 13, no. 1, pp. 54–65, 2022, doi: 10.24002/jbi.v13i1.5022.
 - [5] A. Hosna, E. Merry, J. Gyalmo, Z. Alom, Z. Aung, and M. A. Azim, "Transfer learning: a friendly introduction," *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00652-w.
 - [6] A. Fuadi and A. Suharso, "Perbandingan Arsitektur Mobilenet Dan Nasnetmobile Untuk Klasifikasi Penyakit Pada Citra Daun Kentang," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 7, no. 3, pp. 701–710, 2022, doi: 10.29100/jipi.v7i3.3026.
 - [7] S. Suprihanto, I. Awaludin, M. Fadhil, and M. A. Z. Zulfikor, "Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta," *J. Inform.*, vol. 9, no. 2, pp. 116–122, 2022, doi: 10.31294/inf.v9i1.13049.
 - [8] P. A. P. Huda, A. A. Riadi, and Evanita, "Klasifikasi Penyakit Tanaman Pada Daun Apel Dan Anggur Menggunakan Convolutional Neural Networks," *JUMIKA J. Manaj. Inform.*, vol. 8, no. 1, pp. 10–17, 2021.
 - [9] S. H. Wang and Y. D. Zhang, "DenseNet-201-Based Deep Neural Network with Composite Learning Factor and Precomputation for Multiple Sclerosis Classification," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 16, no. 2s, 2020, doi: 10.1145/3341095.
 - [10] M. Ilhamsyah and U. Enri, "Identification of Bacterial Spot Diseases on Paprika Leaves Using Cnn and Transfer Learning," *J. Pilar Nusa Mandiri*, vol. 18, no. 1, pp. 17–24, 2022, doi: 10.33480/pilar.v18i1.2755.
 - [11] M. T. Islam, B. M. N. Karim Siddique, S. Rahman, and T. Jabid, "Image Recognition with Deep Learning," *2018 Int. Conf. Intell. Informatics Biomed. Sci. ICIIBMS 2018*, vol. 3, pp. 106–110, 2018, doi: 10.1109/ICIIBMS.2018.8550021.
 - [12] F. Salim, F. Saeed, S. Basurra, S. N. Qasem, and T. Al-Hadhrami, "DenseNet-201 and Xception Pre-Trained Deep Learning Models for Fruit Recognition," *Electron.*, vol. 12, no. 14, 2023, doi: 10.3390/electronics12143132.
 - [13] A. P. Syahputra, A. C. Siregar, and R. W. S. Insani, "Comparison of CNN Models With Transfer Learning in the Classification of Insect Pests," *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, vol. 17, no. 1, p. 103, 2023, doi: 10.22146/ijccs.80956.
 - [14] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, 2021, doi: 10.1109/JPROC.2020.3004555.
 - [15] Putri Teresia Ompusunggu, "Klasifikasi Penyakit Tanaman Pada Daun Kentang Dengan Metode Convolutional Neural Network Arsitektur Mobilenet," *J. Syntax FUSION*, vol. 2, no. 8.5.2017, pp. 2003–2005, 2022.
 - [16] R. Rismiyati and A. Luthfiarta, "VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification," *Telematika*, vol. 18, no. 1, p. 37, 2021, doi: 10.31315/telematika.v18i1.4025.

-
- [17] N. E. M. Khalifa, M. Loey, and M. H. N. Taha, "Insect pests recognition based on deep transfer learning models," *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 1, pp. 60–68, 2020.