

Analisis Performa Deteksi Penyakit Padi Dengan Model Klasifikasi Gambar Menggunakan Teachable Machine

Ilham Hidayatullah¹, Ali Nugroho Septihadi², Fredy Susanto³, M. Bucci Ryando⁴

^{1,2,3,4}Teknik Informatika, Institut Teknologi dan Bisnis Bina Sarana Global, Indonesia

Email: ¹1120120001@global.ac.id, ²1120120131@global.ac.id, ³fredysusanto@global.ac.id, ⁴bucci@global.ac.id

Abstrak - Penyakit padi menjadi momok bagi petani karena berpotensi menurunkan hasil panen drastis. Deteksi dini penyakit padi krusial untuk menekan kerugian ekonomi. Penelitian ini menganalisis performa model klasifikasi gambar dalam mendeteksi penyakit padi memanfaatkan Teachable Machine. Gambar penyakit padi dikumpulkan dari berbagai sumber dan melalui proses prapengolahan untuk meningkatkan kualitas. Model klasifikasi gambar dibangun menggunakan Teachable Machine dengan arsitektur Convolutional Neural Network (CNN). Hasil penelitian menunjukkan akurasi model klasifikasi gambar sangat memuaskan, mencapai 99%. Model ini efektif dalam mendeteksi penyakit blas, hawar daun bakteri, dan tungau padi. Penelitian ini membuktikan bahwa Teachable Machine berpotensi sebagai alat efektif untuk mendeteksi penyakit padi dengan akurasi tinggi. Dengan demikian, petani dapat melakukan deteksi dini penyakit padi sehingga memungkinkan tindakan pencegahan dan pengendalian yang tepat.

Kata Kunci - Pembelajaran Mesin, Akurasi Model, Teachable Machine, Deteksi Penyakit

Abstract - Rice diseases are a threat to farmers because they have the potential to drastically reduce crop yields. Early detection of rice diseases is crucial to reduce economic losses. This study analyzes the performance of image classification models in detecting rice diseases using Teachable Machine. Rice disease images were collected from various sources and went through a pre-processing process to improve quality. The image classification model was built using Teachable Machine with Convolutional Neural Network (CNN) architecture. The results showed that the accuracy of the image classification model was very satisfactory, reaching 99%. This model is effective in detecting blast disease, bacterial leaf blight, and rice mites. This study proves that Teachable Machine has the potential to be an effective tool for detecting rice diseases with high accuracy. Thus, farmers can carry out early detection of rice diseases so that appropriate preventive and control measures can be taken.

Keywords - Machine Learning, Model Accuracy, Teachable Machine, Disease Detection

I. PENDAHULUAN

Pertumbuhan dan produksi tanaman padi memainkan peran krusial dalam memenuhi kebutuhan pangan suatu negara.[1][2] Namun, tantangan utama yang dihadapi petani adalah serangan penyakit tanaman yang dapat menyebabkan

penurunan signifikan dalam hasil panen jika tidak diidentifikasi dan ditangani dengan tepat serta cepat.[3] Penyakit-penyakit seperti blas, hawar daun, karat, dan berbagai penyakit virus lainnya menjadi ancaman serius bagi tanaman padi.[4][5] Kepekaan tanaman padi terhadap penyakit ini memungkinkan penyebarannya dengan cepat, yang pada gilirannya dapat mengakibatkan kerugian ekonomi yang besar bagi petani dan bahkan mengancam keamanan pangan global.[6]

Dalam era teknologi informasi saat ini, penggunaan teknologi pengolahan citra dan kecerdasan buatan telah menawarkan solusi yang potensial untuk mengatasi masalah ini.[7] Salah satu metode yang muncul adalah penggunaan model klasifikasi gambar yang mampu mengidentifikasi penyakit tanaman padi dengan akurasi tinggi.[8][9] Teachable Machine, sebagai platform yang memungkinkan pengguna untuk melatih model machine learning tanpa perlu pengetahuan mendalam tentang pemrograman atau kecerdasan buatan, menawarkan kemudahan dalam implementasi solusi ini.[10] Penerapan Teachable Machine dalam analisis deteksi penyakit padi menjanjikan pendekatan yang inovatif dan dapat diakses bagi para petani dan peneliti di lapangan dengan memanfaatkan teknologi ini.[11][12]

Penelitian ini bertujuan untuk melakukan analisis terhadap performa deteksi penyakit pada tanaman padi menggunakan model klasifikasi gambar yang dibangun dengan Teachable Machine. Dengan memanfaatkan teknologi ini, diharapkan dapat diperoleh pemahaman mendalam mengenai efektivitas dan akurasi sistem deteksi yang dikembangkan dalam mendukung keberhasilan pengelolaan kesehatan tanaman padi secara praktis dan efisien.

Penelitian sebelumnya mengenai analisa performa deteksi penyakit padi dengan *Convolutional Neural Network* (CNN) sudah dilakukan dengan judul “Deteksi Penyakit pada Tanaman Padi Menggunakan Pengolahan Citra Digital dengan Metode Convolutional Neural Network”[13] Dalam penelitian ini, dataset yang digunakan terdiri dari 16000 citra yang mencakup tiga kelas penyakit (bacterial leaf blight, brown spot, leaf smut) dan satu kelas tanaman padi sehat. Dataset ini diambil dari www.kaggle.net serta penelitian sebelumnya, yang memastikan keberagaman dan kualitas data. Penelitian ini mengevaluasi performa model CNN dengan menguji parameter seperti jumlah hidden layer dan jenis optimizer. Hasil penelitian menunjukkan bahwa penggunaan empat hidden layer dan optimizer Adam menghasilkan performa terbaik dengan akurasi sebesar

99,66%, precision, recall, dan F1-score sebesar 100%, serta loss sebesar 0,0047. Hasil penelitian ini juga menunjukkan bahwa metode ini dapat mencapai akurasi yang sangat tinggi, yang menunjukkan potensi besar untuk diterapkan dalam skala yang lebih luas di bidang pertanian. Dengan perkembangan teknologi yang terus berlangsung, diharapkan pendekatan ini dapat semakin disempurnakan dan diimplementasikan secara luas untuk meningkatkan produktivitas dan ketahanan pangan global.

Penelitian kedua sebelumnya mengenai analisa performa deteksi penyakit daun teh dengan metode *Machine Learning* sudah dilakukan dengan judul “Analisis Kinerja Algoritma Machine Learning Untuk Deteksi Penyakit Daun Teh Dengan Particle Swarm Optimization”[14] Penelitian ini menunjukkan bahwa optimasi dengan (PSO) meningkatkan akurasi algoritma Machine Learning dalam mendeteksi penyakit daun teh. Akurasi Support Vector Machine (SVM) standar sebesar 87% meningkat menjadi 91.68% setelah optimasi. Hal serupa terjadi pada Gradient Boosting, di mana akurasinya meningkat dari 89% menjadi 91% setelah dioptimasi dengan PSO.

Peningkatan akurasi ini menunjukkan bahwa optimasi PSO mampu memaksimalkan kinerja algoritma Machine Learning dalam mendeteksi penyakit daun teh. Diharapkan penelitian ini dapat membantu petani teh dalam mendeteksi penyakit secara cepat dan meminimalisir peran ahli.

Penelitian ketiga sebelumnya mengenai performa klasifikasi penyakit padi dengan metode *Transfer Learning* sudah dilakukan dengan judul “Fine-Tuning Model Transfer Learning VGG16 Untuk Klasifikasi Citra Penyakit Tanaman Padi”[15] Penelitian ini berfokus pada pengembangan model klasifikasi penyakit tanaman padi menggunakan Convolutional Neural Network (CNN) dengan arsitektur VGG16 dan teknik fine-tuning. Tujuan penelitian ini adalah untuk menganalisis kinerja model dalam mengklasifikasikan empat jenis penyakit padi, yaitu brownspoty, healthy, hispa, dan leafblast. Hasil penelitian menunjukkan bahwa VGG16 dengan fine-tuning menghasilkan akurasi yang lebih tinggi dibandingkan dengan VGG16 tanpa fine-tuning. Pada proses training, VGG16 dengan fine-tuning mencapai akurasi 63.50%, sedangkan VGG16 tanpa fine-tuning hanya mencapai 50.88%. Hal serupa terjadi pada proses validasi dan testing, di mana VGG16 dengan fine-tuning menunjukkan performa yang lebih baik.

Penelitian ini menunjukkan bahwa teknik fine-tuning dapat meningkatkan akurasi model CNN VGG16 dalam mengklasifikasikan penyakit tanaman padi. Penggunaan model ini dapat membantu petani dalam mendeteksi penyakit padi secara dini dan tepat, sehingga dapat meningkatkan hasil panen dan kualitas beras. Penelitian ini memberikan kontribusi penting dalam bidang klasifikasi penyakit tanaman padi. Penggunaan CNN VGG16 dengan fine-tuning menunjukkan potensi yang menjanjikan untuk meningkatkan akurasi deteksi penyakit.

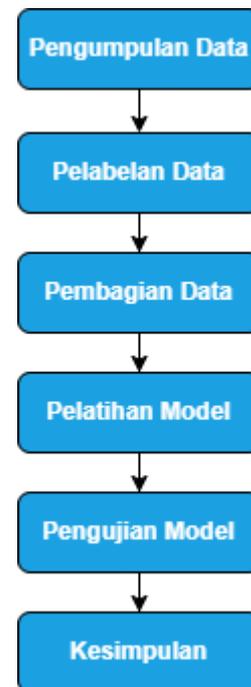
Penelitian kami ini memiliki kebaruan dibandingkan penelitian sebelumnya dengan memanfaatkan Teachable Machine, sebuah platform pembelajaran mesin yang dapat

digunakan tanpa keahlian pemrograman yang mendalam. Pendekatan ini memberikan keunggulan berupa kemudahan implementasi dan aksesibilitas yang lebih baik bagi petani dan peneliti lapangan. Berbeda dengan penelitian sebelumnya yang menggunakan metode CNN atau fine-tuning transfer learning, Teachable Machine menawarkan solusi praktis untuk deteksi penyakit padi secara cepat dan efisien, tanpa memerlukan perangkat keras khusus atau algoritma yang kompleks.

II. METODE PENELITIAN

A. Diagram Alur Penelitian

Penelitian ini dimulai dengan tahap pengumpulan data, di mana data-data yang relevan dan diperlukan dikumpulkan dari website <https://www.kaggle.com>. Setelah data terkumpul, langkah selanjutnya adalah pelabelan data, yaitu proses memberikan klasifikasi atau label pada setiap data sesuai dengan tujuan penelitian. Setelah data dilabeli, tahap berikutnya adalah pembagian data, di mana data dibagi menjadi subset untuk keperluan pelatihan model dan pengujian. Setelah pembagian data selesai, dilakukan pelatihan model menggunakan data pelatihan yang telah disiapkan sebelumnya. Setelah model terlatih, langkah terakhir adalah pengujian model, di mana performa model dievaluasi menggunakan data uji yang terpisah. Berdasarkan hasil pengujian model, kesimpulan atau temuan dari penelitian dapat ditarik. Diagram alur penelitian ini dapat dilihat pada gambar 1 dibawah ini.



Gambar 1. Diagram Alur Penelitian

B. Pengumpulan Data

Dalam tahap ini, peneliti mengumpulkan data dari sebuah dataset yang tersedia di situs website <https://www.kaggle.com/datasets/nirmalsankalana/rice-leaf-disease-image>. Kaggle sendiri adalah sebuah platform yang menyediakan berbagai dataset untuk keperluan analisis dan penelitian.[16]

C. Pelabelan Data

Langkah selanjutnya adalah pelabelan data. Proses ini mencakup penugasan label atau kategori yang sesuai untuk setiap entitas data dalam dataset. Pelabelan data merupakan tahapan krusial yang memungkinkan model atau algoritma *machine learning* untuk memahami dan mengklasifikasikan data dengan tepat.

D. Pembagian Data

Pada proses ini melibatkan pemisahan dataset menjadi subset yang berbeda untuk keperluan pelatihan, dan pengujian model. Data pelatihan digunakan untuk melatih model *machine learning* atau statistik, dan data pengujian digunakan untuk menguji performa model secara independen terhadap data baru yang tidak digunakan dalam pelatihan.

E. Pelatihan Model

Pelatihan model merupakan salah satu tahapan kunci dalam penelitian ini.[17] Proses ini melibatkan penggunaan dataset yang telah dipersiapkan untuk mengajarkan model *machine learning*. Selama pelatihan, model disesuaikan dengan menggunakan algoritma pembelajaran sesuai dengan platform yang digunakan. Hyperparameter model seperti jumlah epoch, batch size, dan learning rate dapat disesuaikan untuk mengoptimalkan performa model. Hasil dari pelatihan ini adalah model yang terlatih, yang diharapkan mampu memberikan prediksi atau analisis yang akurat saat diterapkan pada data baru atau situasi yang serupa.

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Dari banyaknya data yang di dapatkan sebelumnya, peneliti hanya mengambil 5200 gambar yang di bagi menjadi 1300 per setiap kelas *Bacterialblight*, *Blast*, *Brownspot*, dan *Tungro*.



Gambar 2. Sampel Gambar *Bacterial Blight*



Gambar 3. Sampel Gambar *Bacterial Blast*



Gambar 4. Sampel Gambar *Bacterial Brownspot*



Gambar 5. Sampel Gambar *Bacterial Tungro*

B. Pelabelan Data

Pelabelan data dalam penelitian ini dilakukan dengan mengidentifikasi empat jenis utama penyakit pada tanaman padi, yaitu *Bacterial Blight*, *Blast*, *Brown Spot*, dan *Tungro*. Setiap gambar dalam dataset diberi label sesuai dengan jenis penyakit yang terdapat pada daun padi yang ditampilkan. Proses ini penting untuk memastikan bahwa model klasifikasi gambar yang dikembangkan dapat mengenali dan membedakan dengan tepat setiap jenis penyakit tersebut. Dengan demikian, penggunaan label yang akurat memungkinkan model untuk belajar dengan efektif dalam mengidentifikasi dan mendiagnosis masalah kesehatan tanaman secara spesifik.

Tabel 1. Pelabelan Kelas

No	Label
1	<i>Bacterial Blight</i>
2	<i>Blast</i>
3	<i>Brown Spot</i>
4	<i>Tungro</i>

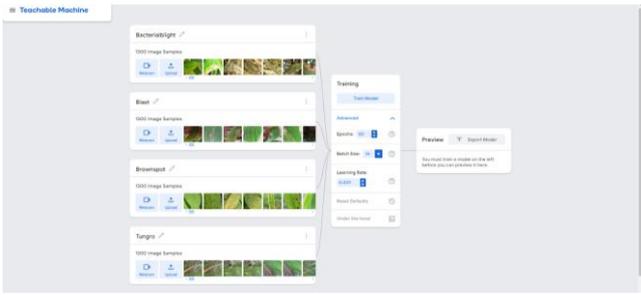
C. Pembagian Data

Platform Teachable Machine memfasilitasi pembagian data secara otomatis dengan perbandingan umum 80:20 untuk pelatihan dan pengujian model. Dalam konteks ini, 80% dari data digunakan untuk melatih model dalam mengenali pola dan fitur yang terkait dengan penyakit pada daun padi. Sementara itu, 20% sisanya dialokasikan untuk pengujian, memastikan bahwa model memiliki kinerja yang baik dalam memprediksi penyakit pada data yang belum pernah dilihat sebelumnya.

D. Pelatihan Model

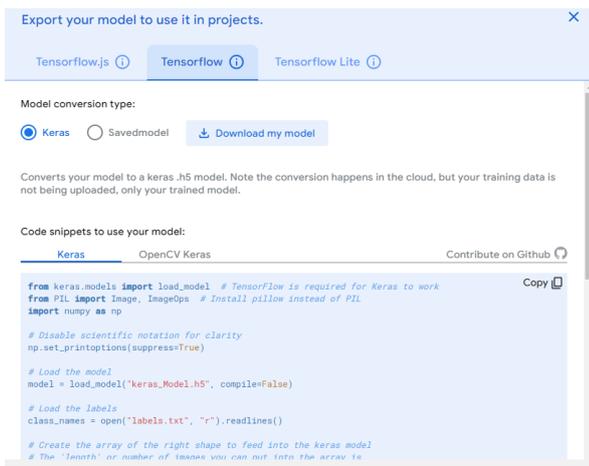
Untuk pelatihan model dalam penelitian ini, peneliti menggunakan platform *Teachable Machine* dari Google, kami mengikuti pengaturan *hyperparameter* default yang telah disediakan. Proses pelatihan dilakukan dengan *epoch* 50, dimana setiap epoch mewakili satu kali proses melalui seluruh dataset pelatihan. Untuk *Batch size* 16, yang menandakan jumlah sampel data yang diproses dalam setiap iterasi pelatihan untuk memperbarui bobot model. *Learning rate* yang digunakan adalah 0.01, yang menentukan seberapa besar perubahan yang diberikan pada bobot model berdasarkan hasil dari fungsi loss.

Penggunaan pengaturan default ini memastikan bahwa model yang kami kembangkan menjalani pelatihan yang konsisten dengan praktik terbaik yang direkomendasikan oleh platform *Teachable Machine*. Dengan demikian, kami berharap dapat menghasilkan model yang optimal dan dapat diandalkan untuk mencapai tujuan penelitian kami dengan akurasi dan keandalan yang tinggi.



Gambar 6. Pelatihan Model Dengan Teachable Machine

Setelah model dilatih menggunakan Teachable Machine, langkah selanjutnya adalah mengonversi model tersebut ke dalam bentuk model Keras untuk melakukan pengujian.



Gambar 7. Export Model Kedalam Format Keras

E. Pengujian Model

Selanjutnya model dimpor ke Google Colab untuk melakukan pengujian. Pada tahap ini, peneliti menyusun beberapa sampel gambar yang relevan untuk diuji terdiri dari beberapa gambar yang sudah dilatih dan ada juga gambar yang belum dilatih untuk memastikan bahwa model bekerja dengan baik andaipun menggunakan sampel gambar yang belum dilatih sama sekali, kemudian selanjutnya adalah memasukkannya ke dalam folder yang telah dipersiapkan dengan label-label yang sesuai. Proses ini memastikan bahwa kami dapat menguji performa model

secara komprehensif terhadap dataset yang diatur dengan baik, menggunakan infrastruktur yang kuat yang disediakan oleh platform Google Colab.

Tahap selanjutnya adalah pengujian model, ketika kode dijalankan di Google Colab, hasilnya akan menampilkan prediksi beserta label dari gambar yang diuji. Contoh di bawah ini menunjukkan pengujian terhadap gambar yang dikategorikan sebagai "Bacterialblight", dengan hasil prediksi sebesar 1.0. Pengujian ini dilakukan secara berulang hingga 20 kali, di mana setiap kelas akan diuji menggunakan gambar yang tidak termasuk dalam data yang dilatih terlebih dahulu.

```

from keras.models import load_model # TensorFlow is required for Keras to work
from PIL import Image, ImageOps # Install pillow instead of PIL
import numpy as np

# Disable scientific notation for clarity
np.set_printoptions(suppress=True)

# Load the model
model = load_model("keras_model.h5", compile=False)

# Load the labels
class_names = open("labels.txt", "r").readlines()

# Create the array of the right shape to feed into the keras model
# The 'length' or number of images you can put into the array is
# determined by the first position in the shape tuple, in this case 1
data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)

# Replace this with the path to your image
image = Image.open("Bacterialblight/BACTERIALBLIGHT1_241.JPG").convert("RGB")

# resizing the image to be at least 224x224 and then cropping from the center
size = (224, 224)
image = ImageOps.fit(image, size, Image.Resampling.LANCZOS)

# turn the image into a numpy array
image_array = np.asarray(image)

# Normalize the image
normalized_image_array = (image_array.astype(np.float32) / 127.5) - 1

# Load the image into the array
data[0] = normalized_image_array

# Predicts the model
prediction = model.predict(data)
index = np.argmax(prediction)
class_name = class_names[index]
confidence_score = prediction[0][index]

# Print prediction and confidence score
print("Class:", class_name[2:], end="")
print("Confidence Score:", confidence_score)
    
```

1/1 [=====] - 1s 1s/step
 Class: Bacterialbli...
 Confidence Score: 1.0

Gambar 8. Pengujian dengan Google Colab

Tabel 2. Hasil Pengujian Model

No	Nama Gambar	Kategori Sebenarnya	Sampel Gambar Train/Test	Prediksi	Persentase Prediksi
1	BACTERAILBLIGHT3_003.jpg	Bacterial Blight	Train	Bacterial Blight	1.0
2	BACTERAILBLIGHT3_023.jpg	Bacterial Blight	Train	Bacterial Blight	1.0
3	BACTERAILBLIGHT3_043.jpg	Bacterial Blight	Train	Bacterial Blight	0.99998546
4	BACTERIALBLIGHT2_212.jpg	Bacterial Blight	Test	Bacterial Blight	1.0
5	BACTERIALBLIGHT2_239.JPG	Bacterial Blight	Test	Bacterial Blight	1.0
6	BLAST1_003.jpg	Blast	Train	Blast	0.9999982
7	BLAST1_043.JPG	Blast	Train	Blast	0.9998982
8	BLAST7_074.jpg	Blast	Train	Blast	0.9934488
9	BLAST9_155.jpg	Blast	Test	Blast	0.7174515
10	BLAST9_156.jpg	Blast	Test	Blast	0.99999166
11	brownspot_orig_010.jpg	Brown Spot	Train	Brown Spot	1.0

12	<i>brownspot_orig_030.jpg</i>	<i>Brown Spot</i>	<i>Train</i>	<i>Brown Spot</i>	1.0
13	<i>brownspot_orig_057.jpg</i>	<i>Brown Spot</i>	<i>Train</i>	<i>Brown Spot</i>	1.0
14	<i>BROWNSPOT7_194.jpg</i>	<i>Brown Spot</i>	<i>Test</i>	<i>Brown Spot</i>	1.0
15	<i>BROWNSPOT7_197.jpg</i>	<i>Brown Spot</i>	<i>Test</i>	<i>Brown Spot</i>	1.0
16	<i>TUNGRO1_001.jpg</i>	<i>Tungro</i>	<i>Train</i>	<i>Tungro</i>	1.0
17	<i>TUNGRO1_044.jpg</i>	<i>Tungro</i>	<i>Train</i>	<i>Tungro</i>	1.0
18	<i>TUNGRO1_069.jpg</i>	<i>Tungro</i>	<i>Train</i>	<i>Tungro</i>	1.0
19	<i>TUNGRO5_196.JPG</i>	<i>Tungro</i>	<i>Test</i>	<i>Tungro</i>	0.9983151
20	<i>TUNGRO5_200.JPG</i>	<i>Tungro</i>	<i>Test</i>	<i>Tungro</i>	1.0

Berdasarkan hasil pengujian yang telah dilakukan, model yang dikembangkan dalam penelitian ini mencapai tingkat akurasi 99% pada dataset pelatihan dan pengujian. Angka ini sebanding dengan penelitian sebelumnya oleh Santosa dkk [13], yang menggunakan metode CNN dengan akurasi 99,66%, serta lebih tinggi dibandingkan penelitian oleh Noprisson [15], yang menggunakan transfer learning VGG16 dengan akurasi 63,5%. Selain itu, pendekatan optimasi PSO pada SVM oleh Rahman, dkk. [14] menghasilkan akurasi maksimum sebesar 91,68%. Hasil ini menunjukkan bahwa pendekatan Teachable Machine memberikan akurasi kompetitif, dengan keunggulan berupa kesederhanaan proses pelatihan dan implementasi.

Dari semua rangkaian uji yang telah dilakukan, hasil ini menunjukkan bahwa *Teachable Machine* memiliki potensi sebagai alat yang efektif untuk mendeteksi penyakit padi dengan akurasi tinggi. Hal ini membuka peluang bagi para petani untuk melakukan deteksi dini penyakit padi, sehingga memungkinkan mereka untuk mengambil tindakan pencegahan dan pengendalian yang tepat. Deteksi dini yang akurat dapat membantu meminimalkan kerugian akibat penyakit padi dan meningkatkan hasil panen, sehingga menunjang ketahanan pangan nasional.

IV. KESIMPULAN DAN SARAN

Hasil pengujian menunjukkan bahwa model yang telah dilatih mampu melakukan klasifikasi gambar dengan tingkat akurasi yang tinggi pada dataset pelatihan (99%). Namun, terdapat variasi dalam tingkat kepercayaan prediksi saat menghadapi dataset uji, terutama pada beberapa contoh dalam kategori Blast, yang menunjukkan perlunya evaluasi lebih lanjut terhadap model dan penambahan variasi data uji untuk meningkatkan kemampuan generalisasi. Penelitian ini memberikan kontribusi yang bermanfaat bagi pertanian modern dengan kemampuan mendeteksi penyakit padi secara akurat, membantu petani mengambil tindakan pencegahan tepat waktu untuk mengurangi kerugian hasil panen dan meningkatkan produktivitas. Untuk penelitian selanjutnya, disarankan untuk memperluas diversifikasi dataset uji serta melakukan evaluasi hyperparameter seperti epoch, batch size, dan learning rate untuk meningkatkan performa prediksi.

DAFTAR PUSTAKA

[1] M. A. R. Siregar, "Peningkatan Produktivitas Tanaman Padi Melalui Penerapan Teknologi Pertanian Terkini," *J. Agribisnis*, vol. 1, no. 1, pp. 1–11, 2023.

[2] S. Wijayanto and M. Yoka Fathoni, "Pengelompokan Produktivitas Tanaman Padi di Jawa Tengah Menggunakan Metode Clustering K-Means," *Jupiter*, vol. 13, no. 2, pp. 212–219, 2021.

[3] E. Anggiratih, S. Siswanti, S. K. Octaviani, and A. Sari, "Klasifikasi Penyakit Tanaman Padi Menggunakan Model Deep Learning Efficientnet B3 dengan Transfer Learning," *J. Ilm. SINUS*, vol. 19, no. 1, p. 75, 2021, doi: 10.30646/sinus.v19i1.526.

[4] P. Purwadi and A. H. Nasyuha, "Implementasi Teorema Bayes Untuk Diagnosa Penyakit Hawar Daun Bakteri (Kresek) Dan Penyakit Blas Tanaman Padi," *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 4, p. 777, 2022, doi: 10.30865/jurikom.v9i4.4350.

[5] A. Jinan, B. H. Hayadi, and U. P. Utama, "Klasifikasi Penyakit Tanaman Padi Menggunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron)," *J. Comput. Eng. Sci.*, vol. 1, no. 2, pp. 37–44, 2022.

[6] Y. Fatman, "Implementasi Forward Chaining Pada Sistem Pakar Sebagai Basis Informasi Persebaran Penyakit Padi," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 3, pp. 1581–1595, 2021, doi: 10.35957/jatisi.v8i3.567.

[7] Ulfah Nur Oktaviana, Ricky Hendrawan, Alfian Dwi Khoirul Annas, and Galih Wasis Wicaksono, "Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model Terlatih Resnet101," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1216–1222, 2021, doi: 10.29207/resti.v5i6.3607.

[8] G. Latif, S. E. Abdelhamid, R. E. Mallouhy, J. Alghazo, and Z. A. Kazimi, "Deep Learning Utilization in Agriculture: Detection of Rice Plant Diseases Using an Improved CNN Model," *Plants*, vol. 11, no. 17, 2022, doi: 10.3390/plants11172230.

[9] S. K. Upadhyay and A. Kumar, "A novel approach for rice plant diseases classification with deep convolutional neural network," *Int. J. Inf. Technol.*, vol. 14, no. 1, pp. 185–199, 2022, doi: 10.1007/s41870-021-00817-5.

[10] N. Malleswari, P. Yogendra Prasad, D. Prasad, M. N. Shetty, and N. Gupta, "Implementation of Machine Learning Based Google Teachable

Machine in Early Childhood Education,” *Artic. Int. J. Early Child. Spec. Educ.*, vol. 14, no. May, p. 2022, 2022, [Online]. Available: <https://www.researchgate.net/publication/360438764>

- [11] C. Chazar and M. H. Rafsanjani, “Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman,” *Pros. Semin. Nas. Inov. dan Adopsi Teknol.*, vol. 2, no. 1, pp. 32–40, 2022, doi: 10.35969/inotek.v2i1.207.
- [12] E. A. U. Malahina, R. P. Hadjon, and F. Y. Bisilisin, “Teachable Machine: Real-Time Attendance of Students Based on Open Source System,” *IJICS (International J. Informatics Comput. Sci.)*, vol. 6, no. 3, p. 140, 2022, doi: 10.30865/ijics.v6i3.4928.
- [13] A. A. Santosa, R. Y. N. Fu’adah, and S. Rizal, “Deteksi Penyakit pada Tanaman Padi Menggunakan Pengolahan Citra Digital dengan Metode Convolutional Neural Network,” *J. Electr. Syst. Control Eng.*, vol. 6, no. 2, pp. 98–108, 2023, doi: 10.31289/jesce.v6i2.7930.
- [14] A. Rahman Ramli and A. Khalil Gibran, “Analisis Kinerja Algoritma Machine Learning Untuk Deteksi Penyakit Daun Teh Dengan Particle Swarm Optimization,” *Essi*, vol. 4, no. November, pp. 97–106, 2023, [Online]. Available: <https://ojs.unm.ac.id/JESSI/index>
- [15] H. Noprisson, “Fine-Tuning Model Transfer Learning VGG16 Untuk Klasifikasi Citra Penyakit Tanaman Padi,” *JSAI (Journal Sci. Appl. Informatics)*, vol. 5, no. 3, pp. 244–249, 2022, doi: 10.36085/jsai.v5i3.3609.
- [16] A. M. Ibrahim, M. Elbasheir, S. Badawi, A. Mohammed, and A. F. M. Alalmin, “Skin Cancer Classification Using Transfer Learning by VGG16 Architecture (Case Study on Kaggle Dataset),” *J. Intell. Learn. Syst. Appl.*, vol. 15, no. 03, pp. 67–75, 2023, doi: 10.4236/jilsa.2023.153005.
- [17] N. Takano and G. Alaghband, “SRGAN: Training Dataset Matters,” pp. 1–7, 2019, [Online]. Available: <http://arxiv.org/abs/1903.09922>