

Deteksi Penyakit pada Tanaman Padi Menggunakan *MobileNet* Transfer *Learning* Berbasis Android

Herwina^{1*}, Darmatasia², Ashabul Kahfi Ash Shiddiq³, Theddy Dzikrullah Syahputra⁴

Universitas Islam Negeri Alauddin Makassar¹

herwinawhyna@gmail.com^{1*}, darmatasia@uin-alauddin.ac.id², ashabulratakan@gmail.com³,
theddyDzikrullah@gmail.com⁴,

Abstrak

Beras merupakan makanan pokok di beberapa negara termasuk di Indonesia. Untuk menghasilkan beras yang berkualitas, diperlukan pemeliharaan pada tanaman padi mulai dari penanaman hingga panen. Salah satu masalah yang sering dialami oleh petani adalah adanya penyakit yang menyerang tanaman padi. Keterbatasan pengetahuan pada sebagian petani menyebabkan petani tidak memahami kondisi tanamannya sehingga mengalami keterlambatan penanganan pada saat tanaman terserang penyakit. Penelitian ini bertujuan untuk membangun sebuah aplikasi yang dapat mendeteksi penyakit pada tanaman padi yang menyerang daun padi. Jenis penyakit yang akan dideteksi yaitu Leaf Smut, Brown Spot, dan Bacterial Leaf Blight. Pada penelitian ini menggunakan pendekatan transfer learning dengan Algoritma Convolutional Neural Network untuk mendeteksi penyakit pada daun padi. Adapun arsitektur yang digunakan yaitu MobileNetV1 dengan akurasi sebesar 94% dan MobileNetV2 dengan akurasi sebesar 95%. Input citra yang digunakan berukuran 224x224 piksel. Model yang telah dilatih selanjutnya diintegrasikan ke dalam aplikasi berbasis Android. Hasil pengujian pada aplikasi Android menunjukkan bahwa model dapat mendeteksi penyakit pada daun padi. Hasil penelitian diharapkan dapat memudahkan petani dalam mendeteksi kemungkinan serangan penyakit pada tanaman padi sehingga dapat meminimalisir gagal panen.

Kata kunci: Android, MobileNet, Penyakit Daun Padi

Abstract

Rice is a main food in several countries including Indonesia. To produce quality rice, maintenance of paddy is needed from planting to harvesting. One of the common problems that need to be addressed by farmers is diseases that attack the paddy. The limited knowledge of some farmers causes they don't understand the condition of their paddy and it makes improper handling when paddy is attacked by diseases. This study aims to build an application that can detect diseases in paddy plants that attack paddy leaves. The types of diseases that will be detected are Leaf Smut, Brown Spot, dan Bacterial Leaf Blight. This study uses a transfer learning approach with the Convolutional Neural Network Algorithm to detect diseases in paddy leaves. The architecture used is MobileNetV1 with an accuracy of 94% and MobileNetV2 with an accuracy of 95%. The input size of the images is 224x224. The model that has been trained then integrated into an Android-based application. The test results show that the model can detect disease in paddy leaves. The results of this study are expected can help the farmer to detect possible disease attacks on paddy plants to minimize the crop failure.

Keywords: Android, MobileNet, Paddy Leaf Disease

1. PENDAHULUAN

Beras merupakan salah satu makanan pokok terutama di negara-negara Asia. Indonesia sebagai salah satu negara yang terletak di benua Asia merupakan salah satu produsen beras terbesar dunia. Namun demikian, para petani di Indonesia belum mampu memenuhi kebutuhan beras untuk masyarakat Indonesia. Selain

karena jumlah populasi yang terus meningkat, diperlukan juga teknik dan inovasi yang optimal dalam perawatan dan pemeliharaan tanaman padi. Perawatan dan pemeliharaan yang tidak optimal dapat menyebabkan produksi yang menurun. Berbagai upaya telah dilakukan oleh pemerintah untuk mendorong peningkatan produksi beras

seperti penerapan inovasi teknologi pertanian, penyediaan pupuk bersubsidi, dan perbaikan infrastruktur sawah.

Salah satu masalah yang sering menyebabkan terjadinya gagal panen adalah adanya serangan penyakit pada tanaman padi. Penyakit dapat disebabkan oleh hama ataupun bakteri. Penyakit pada tanaman padi umumnya menyerang daun atau batang padi yang jika hal tersebut dibiarkan maka dapat menyebabkan kerugian yang besar. Oleh karena itu diperlukan deteksi yang lebih dini terhadap serangan penyakit pada tanaman padi agar dapat meminimalisir kerugian.

Pengendalian terhadap penyakit ataupun hama yang menyerang tanaman padi tentunya dilakukan oleh petani. Namun keterbatasan pengetahuan dan pengalaman terutama petani pemula dapat menyebabkan penanganan yang tidak sesuai atau keterlambatan penanganan. Salah satu upaya yang dapat dilakukan untuk mengatasi hal tersebut adalah dengan memberikan bekal pengetahuan kepada petani. Namun hal tersebut membutuhkan waktu yang lama, mengingat jumlah petani yang ada di Indonesia cukup banyak.

Seiring dengan perkembangan teknologi, salah satu alternatif yang juga dapat dilakukan untuk mengatasi keterbatasan pengetahuan para petani adalah dengan menyediakan sebuah aplikasi yang dapat membantu petani dalam mendeteksi lebih dini berbagai jenis penyakit yang menyerang tanaman padi. Aplikasi tersebut dapat digunakan oleh semua petani dimana pun dan kapan pun.

Aplikasi deteksi penyakit pada tanaman padi telah banyak dikembangkan. Dalam penelitian [1] menggunakan pendekatan *transfer learning* dengan arsitektur *MobileNetV3* dan *EfficientNet B3* untuk mendeteksi penyakit pada daun padi yang terdiri dari *brown spot* dan *bacterial leaf*. Hasil penelitian menunjukkan bahwa arsitektur *EfficientNet B3* memperoleh nilai akurasi

yang lebih tinggi yaitu 99%, sedangkan *MobileNet V3* hanya memperoleh akurasi sebesar 79.53%.

Deteksi penyakit tanaman padi yang menyerang melalui daun telah diteliti oleh [2]. Metode yang digunakan adalah *deep learning* dengan arsitektur *MobileNetV1*. Jenis penyakit yang dideteksi terdiri dari 3 jenis yaitu *bacterial leaf blight*, *brown spot*, *leaf smut* dengan jumlah data sebanyak 120 gambar. Hasil akurasi yang diperoleh adalah 92%.

Penelitian yang dilakukan oleh [3] dengan menggunakan *Adaptif Neuro Fuzzy Inference System* (ANFIS) untuk mendeteksi 3 jenis penyakit yaitu *bacterial leaf blight*, *brown spot*, *leaf smut* dengan akurasi sebesar 98.5%.

Penelitian oleh [4] menggunakan ekstraksi fitur *Color Histogram* dan *Random Forest* sebagai *classifier* untuk mendeteksi jenis penyakit pada daun padi yang terdiri dari 3 jenis penyakit yaitu *bacterial leaf blight*, *brown spot*, *leaf smut*. Dalam penelitian tersebut dilakukan augmentasi citra untuk menambah jumlah citra asli. Hasil penelitian memperoleh akurasi sebesar 99.65%.

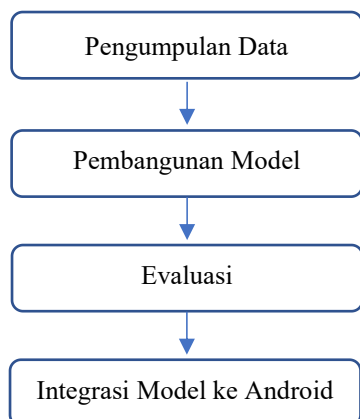
Berdasarkan beberapa penelitian yang telah dilakukan, dapat dilihat bahwa deteksi penyakit padi dapat dilakukan melalui gejala yang tampak pada daun. Pada penelitian ini akan dilakukan deteksi penyakit yang menyerang daun padi. Adapun jenis penyakit yang akan dideteksi terdiri dari 3 jenis yaitu *bacterial leaf blight*, *brown spot*, dan *leaf smut*. Deteksi jenis penyakit yang dilakukan menggunakan pendekatan *transfer learning* dengan arsitektur *MobileNetV1* dan *MobileNetV2*. *Transfer learning* memanfaatkan bobot pada model yang telah dilatih menggunakan dataset tertentu untuk menyelesaikan permasalahan serupa namun dengan dataset yang baru. Model yang telah dilatih selanjutnya akan digunakan pada aplikasi yang berbasis Andorid.

Penelitian ini diharapkan dapat memudahkan petani dalam mendeteksi

lebih dini penyakit pada tanaman padi yang menyerang daun padi agar dapat meningkat produksi dan meminimalisir kerugian.

2. METODE

Secara umum, langkah-langkah penelitian yang dilakukan dapat dilihat pada Gambar 1 berikut:



Gambar 1. Tahapan Penelitian

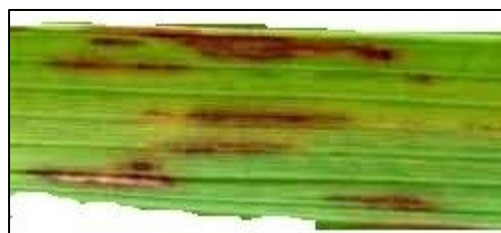
a. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diperoleh dari Kaggle [5]. Dataset terdiri dari 3 bagian yaitu data untuk pelatihan, validasi, dan pengujian dengan rincian sebagai berikut:

Tabel 1. Dataset
Paddy Leaf Disease [5]

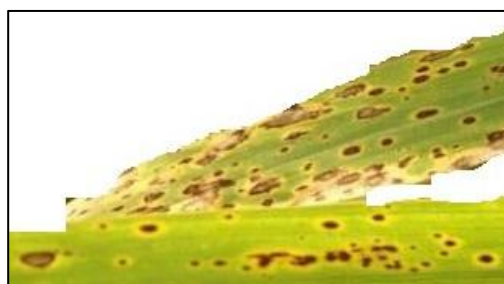
Jenis Penyakit	Data Pelatihan	Data Validasi	Data Pengujian
<i>Leaf Smut</i>	303	77	55
<i>Brown Spot</i>	303	76	54
<i>Bacterial Leaf Blight</i>	305	66	55

Dataset yang digunakan terdiri dari 3 jenis penyakit yaitu *Leaf Smut*, *Brown Spot*, dan *Bacterial Leaf Blight*. *Leaf Smut* adalah salah satu jenis penyakit yang menyerang daun padi yang diakibatkan oleh jamur. Gejala yang tampak pada daun berupa bercak-bercak, kecil, daun menguning dan menggulung.



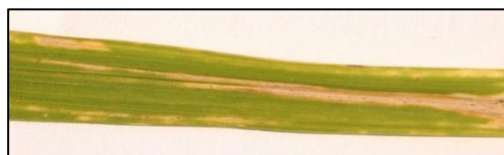
Gambar 2. Contoh Penyakit *Leaf Smut* [5]

Brown Spot merupakan jenis penyakit daun padi yang biasanya mempunyai ciri berupa bercak pada daun padi yang bentuknya khas berbentuk oval dan seragam yang seringkali tersebar di seluruh permukaan daun. Bercak berwarna coklat dilingkari dengan warna abu bagian tengah bercak bulat berwarna putih.



Gambar 3. Contoh Penyakit *Brown Spot* [5]

Bacterial leaf blight merupakan jenis penyakit yang menyerang daun padi disebabkan oleh bakteri. Serangan biasanya dimulai dari tepi daun yang akan menyebabkan daun berwarna hijau kelabu dan mengering, helaian daun akan melengkung diikuti dengan melipatnya helaian daun, dan sepanjang ibu tulang daun berwarna kuning.

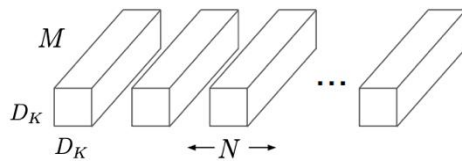


Gambar 4. Contoh Penyakit *Bacterial Leaf Blight* [5]

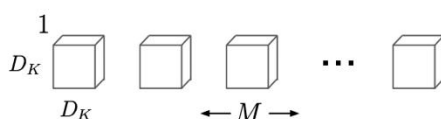
b. Pembangunan Model

Dataset yang telah dikumpulkan selanjutnya digunakan untuk melatih model. Model yang digunakan yaitu *MobileNetV1* dan *MobileNetV2* yang merupakan arsitektur *deep learning*.

MobileNet pada dasarnya merupakan *Convolutional Neural Network* (CNN) yang dirancang untuk aplikasi *mobile* dan *embedded system* yang menggunakan *depthwise separable convolutions*. *Depthwise separable convolutions* terdiri dari dua lapisan, lapisan pertama adalah *depthwise convolution* yang menerapkan filter tunggal pada setiap saluran *input*, sedangkan lapisan kedua adalah *pointwise convolution* yang akan membuat kombinasi linier dari *output* lapisan *depthwise* [6].



Gambar 5. Konvolusi Standar [6]



Gambar 6. Filter *Depthwise Convolutional* [6]

MobileNet mengusulkan global *hyperparameter* baru yang disebut dengan *width multiplier* dan *resolution multiplier*. *Width multiplier* digunakan untuk mengurangi komputasi sedangkan *resolution multiplier* digunakan untuk mengurangi resolusi *input* gambar. *MobileNet* dengan arsitektur demikian dikenal dengan *MobileNetV1*. Adapun arsitektur *MobileNet* dapat dilihat pada Gambar 7.

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5×	Conv dw / s1	$3 \times 3 \times 512$ dw
	Conv / s1	$1 \times 1 \times 512 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Gambar 7. Arsitektur *MobileNet* [6]

Modifikasi versi pertama *MobileNet* diusulkan oleh [7] yang dikenal sebagai *MobileNetV2*. Perubahan utama pada *MobileNetV2* adalah penggunaan *inverted residual blocks* dan *linear bottleneck*. Dalam *bottleneck* terdapat *input* dan *output* antara model sementara lapisan dalam merangkum kemampuan model untuk mengubah *input* dari tingkat yang lebih rendah (yaitu piksel) ke deskriptor dengan tingkat yang lebih tinggi. Semua lapisan kecuali lapisan terakhir dalam arsitektur *MobileNet* diikuti oleh *batchnorm* dan *ReLU nonlinearity* [8].

Penelitian ini menggunakan arsitektur *MobileNet transfer learning* sehingga model tidak dilatih dari awal. *Transfer learning* digunakan untuk meningkatkan *learner* dari satu domain dengan cara mentransfer informasi dari domain serupa.

Transfer learning adalah menggunakan model yang sudah dilatih (*pre-trained*) dengan data latih yang besar kemudian digunakan kembali untuk ekstraksi fitur pada sebuah data baru [9]. Pada penelitian ini menggunakan arsitektur *MobileNet* yang telah dilatih menggunakan data ImageNet kemudian dilakukan *fine-tune* yang bertujuan untuk

menyesuaikan fitur spesifik sesuai data yang diberikan sehingga proses *learning* lebih cepat dan lebih akurat.

c. Evaluasi

Evaluasi bertujuan untuk menguji model yang telah dilatih. Model diuji dengan menggunakan data pengujian yang telah disiapkan sebelumnya. Model dievaluasi menggunakan pengukuran *confusion matrix* akurasi, presisi *recall*, dan *F1-score*.

Confusion matrix adalah salah satu pengukuran performa klasifikasi yang berupa tabel kombinasi dari nilai prediksi dan nilai yang sebenarnya.

Tabel 2. *Confusion Matrix*

Aktual	Prediksi	
	<i>True</i>	<i>False</i>
<i>True</i>	TP	FN
<i>False</i>	FP	TN

True Positive (TP) adalah jumlah *record* positif yang diklasifikasikan sebagai positif. *False* positif (FP) adalah jumlah *record* negatif yang diklasifikasikan sebagai positif. *False* negatif (FN) adalah jumlah *record* positif yang diklasifikasikan sebagai negatif. *True* negatif (TN) adalah jumlah *record* negatif yang diklasifikasikan sebagai negatif.

Akurasi mengukur tingkat keakuratan model yang mengukur rasio prediksi kebenaran (positif dan negatif) dari keseluruhan data dan yang dapat dihitung dengan persamaan 1

$$akurasi = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

Presisi merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Presisi dapat dihitung dengan persamaan 2.

$$precision = \frac{TP}{TP + FP} \quad (2)$$

Recall merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. *Recall* dapat dihitung dengan persamaan 3.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-Score adalah proses kalkulasi evaluasi dalam pencarian informasi yang menggabungkan *recall* dan presisi. Nilai terbaik *F1-Score* adalah 1.0 dan nilai terburuknya adalah 0. Jika *F1-Score* memiliki skor yang tinggi artinya bahwa model klasifikasi yang telah dibangun memiliki presisi dan *recall* yang baik. *F1 score* dihitung dengan persamaan 4.

$$F1 - Score = 2 \times \frac{Recall \times Presisi}{Recall + Presisi} \quad (4)$$

d. Integrasi Model ke Android

Model yang telah dibangun dan dievaluasi selanjutnya diintegrasikan ke Android dengan menggunakan TensorFlow Lite. TensorFlow Lite adalah perangkat lunak yang bersifat *open-source* yang dirancang untuk menjalankan model yang sudah ada sebelumnya di perangkat seluler, *embedded*, dan IoT (*Internet of Things*) [10]. TensorFlow Lite mendukung Android dan iOS. TensorFlow Lite tidak dirancang untuk model pelatihan, sehingga untuk mendapatkan model yang akurat, model harus dilatih terlebih dahulu kemudian dikonversi ke format TensorFlow Lite.

3. HASIL DAN PEMBAHASAN

Penelitian ini menggunakan *MobileNetV1* dan *MobileNetV2* yang telah dilatih sebelumnya dengan dataset dari ImageNet. Secara umum, lapisan awal dari arsitektur akan belajar mengenali objek, lapisan tengah akan mempelajari

bentuk objek, dan lapisan akhir mempelajari fitur khusus dari objek. Dalam pendekatan transfer *learning*, lapisan awal dan tengah menggunakan informasi dari model yang telah dilatih (*pre-trained*), sedangkan lapisan akhir dilatih untuk mendapatkan fitur spesifik dari dataset baru sehingga akan mengekstrak informasi baru yang membedakan antara objek baru dengan objek yang lain.

Ketika menggunakan model yang telah dilatih (*pre-trained*) maka model akan mengesktrak beberapa fitur yang berbeda sehingga parameter perlu disesuaikan sebelum mentransfer pengetahuan. Dalam penelitian ini, pada layer akhir ditambahkan 2 *dense layer* yang terdiri dari 128 dan 64 neuron. Parameter yang digunakan diantaranya yaitu, jumlah *epoch* adalah 35, nilai *learning rate* adalah 0.0001, dan *batch size* yaitu 8.

Penelitian ini membandingkan dua *optimizer* yaitu *Stochastic Gradient Descent* (SGD) dan *Adam optimizer* untuk memperoleh hasil yang terbaik. Untuk menghindari *overfitting*, pada penelitian digunakan *dropout* setelah *dense layer* dengan nilai masing-masing 0.5 dan 0.2. Adapun hasil akurasi yang diperoleh pada penelitian ini dapat dilihat pada Tabel 3.

Tabel 3. Hasil Akurasi

Arsitektur	Optimizer	Akurasi
MobileNetV1	SGD	76%
	Adam	94%
MobileNetV2	SGD	74%
	Adam	95%

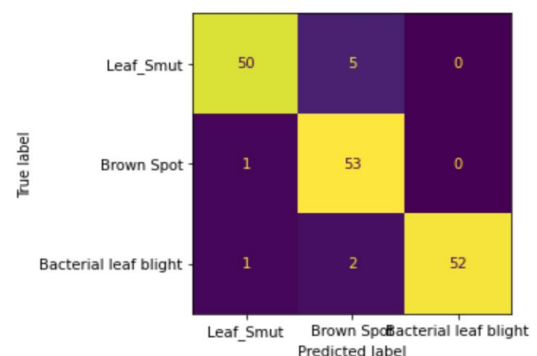
Berdasarkan Tabel 3 dapat dilihat bahwa akurasi terbaik diperoleh dengan arsitektur *MobileNetV2* dengan *optimizer* menggunakan *Adam optimizer*. Secara keseluruhan, tingkat akurasi yang diperoleh mencapai 95% sehingga dapat dikatakan bahwa model *MobileNetV2* dengan transfer *learning fine-tune* memiliki performa yang baik untuk deteksi jenis penyakit pada daun padi.

Adapun grafik hasil akurasi pada setiap *epoch* selama proses pelatihan dapat dilihat pada Gambar 8 berikut:



Gambar 8. Grafik akurasi pelatihan pada setiap *epoch*

Adapun *confusion matrix* dapat dilihat pada Gambar 9. *Confusion matrix* menunjukkan bahwa terdapat 5 data dari kelas *Leaf Smut* yang salah diprediksi sebagai kelas *Brown Spot*, 1 data dari kelas *Brown Spot* yang salah diprediksi sebagai kelas *Leaf Smut*, dan pada kelas *Bacterial Leaf Blight* terdapat 3 data yang salah diprediksi dimana 1 data diprediksi sebagai kelas *Leaf Smut* dan 2 data diprediksi sebagai kelas *Brown Spot*.



Gambar 9. Confusion Matrix

Hasil presisi, *recall*, dan *F1-score* dapat dilihat pada Tabel 4.

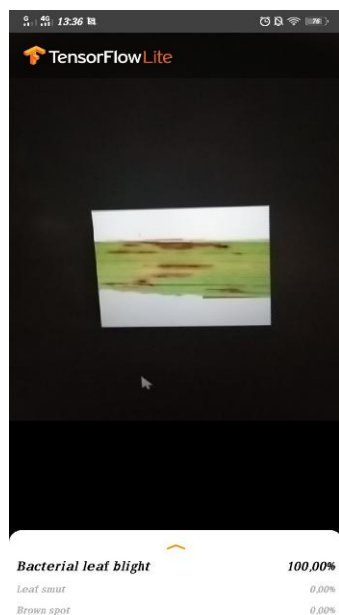
Tabel 4. Nilai Presisi, *Recall*, dan *F1-Score*

Jenis Penyakit	Presisi	Recall	F1-Score
Leaf Smut	96%	91%	93%
Brown Spot	88%	98%	93%

<i>Bacterial Leaf Blight</i>	100%	95%	97%
------------------------------	------	-----	-----

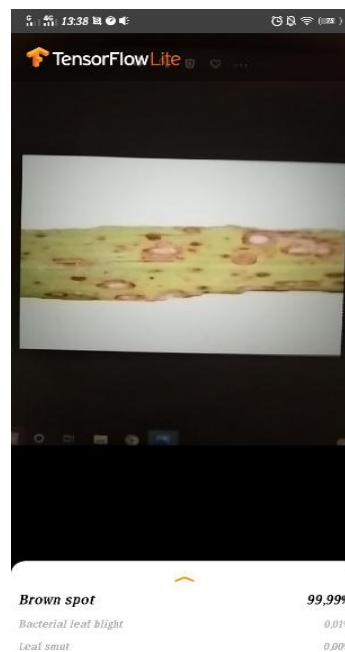
Berdasarkan Tabel 4 dapat dilihat bahwa nilai presisi tertinggi adalah kelas *Bacterial Leaf Blight*, nilai *recall* tertinggi adalah *Brown Spot*, dan F1-Score tertinggi adalah *Bacterial Leaf Blight*. Dengan nilai presisi 100% pada *Bacterial Leaf Blight* menunjukkan bahwa tidak ada data yang salah diprediksi sebagai kelas *Bacterial Leaf Blight*.

Model terbaik yang telah dibangun selanjutnya diintegrasikan ke perangkat berbasis Android dengan menggunakan *framework* Tensorflow Lite. Adapun hasil pengujian pada aplikasi Android dapat dilihat pada Gambar 10, 11, dan 12.



Gambar 10. Hasil pengujian pada penyakit *Bacterial Leaf Blight*

Gambar 10 menunjukkan bahwa aplikasi dapat mendeteksi penyakit padi *Bacterial Leaf Blight* dengan akurasi 100%. Selanjutnya dilakukan pengujian menggunakan gambar daun padi dengan penyakit *Brown Spot*. Adapun hasil pengujiannya dapat dilihat pada Gambar 11.



Gambar 11. Hasil pengujian pada penyakit *Brown Spot*

Pada gambar di atas dilakukan pengujian dengan menggunakan gambar jenis penyakit daun padi *Brown spot* dengan akurasi 99,99%.



Gambar 12. Hasil pengujian pada penyakit *Leaf Smut*

Pada gambar di atas dilakukan pengujian dengan menggunakan gambar

jenis penyakit daun padi *Bacterial Leaf Blight* dengan akurasi 99,80%.

4. KESIMPULAN

Arsitektur MobileNetV2 telah digunakan dalam penelitian ini untuk membangun sebuah model yang dapat mendeteksi jenis penyakit yang menyerang pada daun padi. Model yang telah dibangun telah diintegrasikan ke dalam aplikasi berbasis Android. Hasil penelitian diharapkan dapat digunakan oleh petani untuk memudahkan deteksi dini penyakit yang menyerang pada daun padi.

Secara keseluruhan, hasil penelitian menunjukkan akurasi yang cukup tinggi yaitu 95%. Namun demikian, penelitian ini hanya mampu mendeteksi 3 jenis penyakit yang menyerang pada daun padi. Untuk penelitian selanjutnya dapat dikembangkan agar aplikasi dapat mendeteksi jenis penyakit yang lebih banyak. Selain itu, diharapkan penelitian selanjutnya tidak hanya dapat mendeteksi penyakit yang menyerang pada daun padi tetapi juga penyakit yang menyerang pada batang padi.

5. DAFTAR PUSTAKA

- [1] 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, pp. 75–83, 2021.
- [2] R. A. Saputra, S. Wasyianti, A. Supriyatna, and D. F. Saefudin, "Penerapan Algoritma Convolutional Neural Network Dan Arsitektur MobileNet Pada Aplikasi Deteksi Penyakit Daun Padi," *J. SWABUMI*, vol. 9, no. 2, 2021.
- [3] R. N. Whidhiasih and I. Ekawati, "IDENTIFIKASI JENIS PENYAKIT DAUN PADI MENGGUNAKAN ADAPTIF NEURO FUZZY INFERENE SYSTEM (ANFIS) BERDASARKAN TEKSTUR," in *PROSIDING SEMINAR NASIONAL ENERGI & TEKNOLOGI (SINERGI)*, 2019, pp. 131–140.
- [4] S. Agustiani, Y. T. Arifin, A. Junaidi, S. K. Wildah, and A. Mustopa, "Klasifikasi Penyakit Daun Padi menggunakan Random Forest dan Color Histogram," *J. Komputasi*, vol. 10, no. 1, pp. 65–74, 2022.
- [5] S. AHMED, "paddy leaf disease UCI," 2020. <https://www.kaggle.com/datasets/badhon7432/paddyleafdiseseuci>.
- [6] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." 2017.
- [7] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks." 2019.
- [8] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift." 2015.
- [9] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 1, p. 9, 2016, doi: 10.1186/s40537-016-0043-6.
- [10] "TensorFlow Lite," 2021. <https://www.tensorflow.org/lite/guide>.