

Klasifikasi Tingkat Keparahan Penyakit Leafblast Tanaman Padi Menggunakan MobileNetv2

Imam Fauzi Annur ¹⁾*, Jumhurul Umami ¹⁾, Moch. Nasheh Annafii¹⁾, Niken Trisnaningrum ²⁾ Oddy Virgantara Putra ¹⁾,

Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Darussalam Gontor ^{1,2,3,5)}

Program Studi Agroteknologi, Fakultas Sains dan Teknologi, Universitas Darussalam Gontor ⁴⁾

imamfauziannur@unida.gontor.ac.id ¹⁾*, jumhur@unida.gontor.ac.id ²⁾, nasheh@unida.gontor.ac.id ³⁾, niken.trisnaningrum@unida.gontor.ac.id ⁴⁾, oddy@unida.gontor.ac.id ⁵⁾

Abstrak

Padi merupakan tanaman pangan pokok di Indonesia, dan produksinya merupakan kunci ketahanan pangan negara. Keberhasilan panen merupakan faktor penting dalam pencegahan impor bahan pangan pokok. Tantangan terbesar dalam memanen tanaman adalah adanya virus, jamur, dan hama yang dapat merusak tanaman. Penelitian ini bertujuan untuk membuat sistem klasifikasi tingkat keparahan penyakit daun pada tanaman padi yang terkena penyakit blas daun dengan bantuan algoritma machine learning. MobileNetV2 adalah arsitektur Convolutional Neural Network (CNN) yang menggunakan Depthwise Separable Convolution untuk membangun model yang ringan dan dirancang untuk mengatasi proses yang memiliki resource yang berlebih. Dataset yang digunakan pada penelitian ini merupakan hasil murni observasi peneliti yang sudah divalidasi oleh ahli dengan total 300 data asli. Model MobileNetV2 ternyata sangat berhasil dalam mengklasifikasikan objek, dengan akurasi 78,33%. dengan hasil penelitian ini, petani dapat terbantu dalam mengenali tingkat keparahan penyakit leafblast pada tanaman padi sehingga pemberian bahan kimia berupa fungisida sesuai dengan dosis anjuran tingkat keparahan.

Kata kunci: Klasifikasi, leafblast, padi, citra, model pre-trained, MobileNetV2.

Abstract

[Classification Of Rice Blast Disease Using MobileNetV2] Rice is a staple food crop in Indonesia, and its production is key to the country's food security. Successful harvesting is an important factor in preventing imports of staple foods. The biggest challenge in harvesting crops is the presence of viruses, fungi, and pests that can damage plants. This research aims to create a classification system for leaf disease severity in rice plants affected by leaf blast disease with the help of machine learning algorithms. MobileNetV2 is a Convolutional Neural Network (CNN) architecture that uses Depthwise Separable Convolution to build lightweight models and is designed to overcome processes that have excessive resources. The dataset used in this study is the result of pure researcher observations that have been validated by experts with a total of 300 original data. The MobileNetV2 model turned out to be very successful in classifying objects, with an accuracy of 78.33%. with the results of this study, farmers can be helped in recognizing the severity of leafblast disease in rice plants so that the provision of chemicals in the form of fungicides in accordance with the recommended dose of severity.

Keywords: Classification, leafblast, rice, image, pre-trained model, MobileNetV2

1. PENDAHULUAN

Tanaman padi merupakan salah satu tanaman yang dimana dari zaman dahulu hingga sekarang menjadi penghasil bahan pangan pokok berupa nasi untuk kebanyakan di daerah tropis, terutama di benua Asia dan Afrika. Salah satu negara yang mempunyai kuantitas konsumen terbanyak adalah Indonesia, dimana negara ini merupakan negara Agraris yang menjadikan pertanian merupakan bagian terpenting dalam memegang perairan dan harus dijaga

kestabilannya[1]. Kebutuhan masyarakat akan bahan pangan pokok ini yang dihasilkan oleh tanaman ini terus mengalami peningkatan seiring dengan bertambahnya jumlah penduduk sehingga keberhasilan dalam panen untuk tanaman padi menjadi faktor yang sangat berpengaruh, dimana untuk menghindari impor bahan pangan pokok[2].

Melihat dari keberhasilan panen, permasalahan terbesar yang menyebabkan hasil panen sedikit bahkan menyebabkan kegagalan panen yaitu adanya penyakit

tanaman padi[3] yang sudah terdiagnosis bahkan terlambat untuk ditangani sehingga berakibat sudah menjadi tahap yang parah dan bisa menyebabkan terjadinya suatu kegagalan panen. Penyakit pada tanaman padi ada bermacam macam jenis, salah satunya adalah *leaf blast* atau penyakit blas. Penyakit blas ini disebabkan oleh jamur *Pyricularia Grisea* yang dimana penyakit tersebut mampu menyerang tanaman padi pada berbagai macam stadia pertumbuhan dari benih hingga pertumbuhan pada malai padi (generatif). Pada fase vegetatif, jamur ini dominan menginfeksi bagian pada daun padi sehingga bisa disebut blas padi (*leaf blast*). Sesampainya di fase generatif, selain bisa menginfeksi pada daun juga menginfeksi pada leher malai padi disebut blas leher (*neck blast*)[4]. Dampak penyakit ini bisa menyebabkan batang padi patah dan kematian menyeluruh pada bagian yang terinfeksi.

Mengenai produksi padi diiringi dengan pertambahan penduduk, titik terpenting yang diambil adalah keberhasilan dalam panen. Adanya minimnya pengetahuan petani mengenai tingkat keparahan penyakit pada tanaman padi tentang pengobatan organik dan ramah lingkungan[5]. Oleh karena itu untuk analisa penyakit ditanggulangi dengan pemberian obat seperti fungisida dan pestisida yang dimana sesuai dengan dosis tingkat keparahan penyakit tersebut. Klasifikasi citra pada penyakit tanaman padi berdasarkan tingkat keparahannya dapat membantu petani dalam optimasi pemberian dosis obat yang tepat dan akurat.

MobileNetV2 merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang dimana merupakan penyempurnaan dan pengembangan dari arsitektur *MobileNetV1* dengan melakukan penambahan residual terbalik dengan *linear bottleneck* dan *Shortcut Connections* antar *Bottleneck*[6]. Adapun penggunaan arsitektur *MobileNetV2* ini demi mengatasi proses yang memerlukan *computing resource* yang tinggi untuk menghasilkan efisiensi klasifikasi citra terbaik[7].

Paper ini dibagi menjadi 4 bagian. Pada bagian pertama akan menjelaskan pendahuluan dalam penelitian ini, pada bagian kedua akan menjelaskan bahan dan metode penelitian, pada bagian ketiga menjelaskan hasil dan pembahasan, dan pada bagian yang terakhir terdapat kesimpulan dari peneliti.

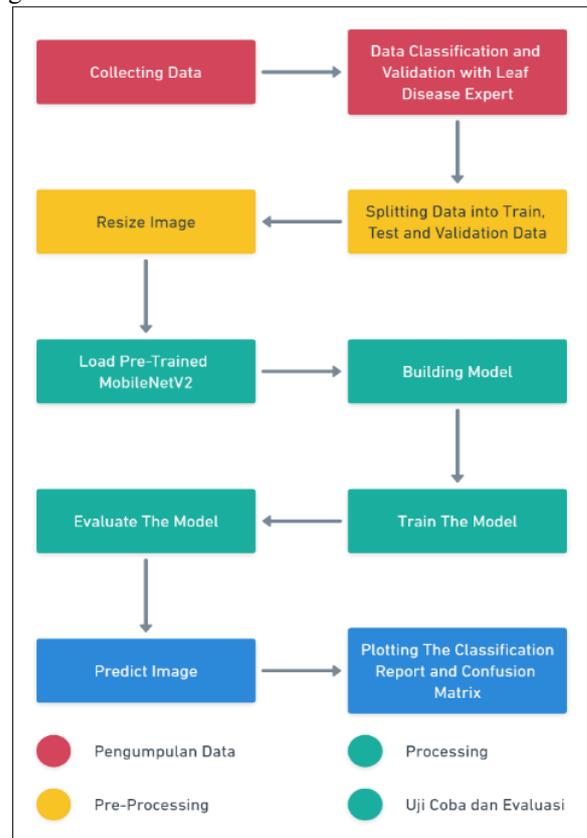
2. BAHAN DAN METODE

Pada penelitian ini, peneliti bertujuan untuk mendapatkan hasil akurasi yang optimal untuk klasifikasi citra tingkat keparahan pada penyakit *leafblast* di tanaman padi menggunakan arsitektur *MobileNetV2*. Arsitektur model ini cocok untuk digunakan klasifikasi suatu citra. Untuk penelitian klasifikasi citra tersebut, peneliti membuat tahapan penelitian yang akan digunakan untuk membangun model klasifikasi tersebut, antara lain sebagai berikut:

1. Tahap Pengumpulan Data
2. Tahap *Pre-Processing*

3. Tahap *Processing*
4. Tahap Uji Coba dan Evaluasi

Adapun tahapan penelitian ini ditampilkan di gambar 1



Gambar 1. Tahapan Penelitian

2.1. Tahap Pengumpulan Data

Pada bagian ini terdapat dua tahapan yang dilakukan dalam penelitian ini. Tahapan pertama adalah pengambilan citra untuk dataset secara observasi. Observasi dilakukan di sawah salah satu anggota dari kelompok “Tani Maju” di kecamatan Babadan, Ponorogo, Jawa Timur. Pengambilan data dilaksanakan pada awal bulan Juli tahun 2022 dengan waktu selama 3 hari berturut-turut pada pagi hari pukul 10.00 – 11.30 dengan cuaca yang cerah dan sedikit berawan. Pengambilan data dilakukan secara langsung di lokasi menggunakan kamera ukuran sebesar 18 MP pada handphone “Redmi 9” dengan setting *ISO* 1000 dan *Saturation* 0,3 serta menggunakan ukuran lensa “Wide”. Hasil dari pengambilan citra secara observasi langsung ke lokasi ditampilkan pada gambar 2. Jumlah data yang diambil untuk dataset sebanyak 300 data citra daun padi. Data yang didapat memiliki berbagai tingkatan keparahan penyakit pada padi. Adapun macam varian tingkat keparahan yang didapat adalah daun sehat-ringan, sedang dan berat.

Setelah melewati proses tahapan pertama, data dikelompokkan berdasarkan kategori tingkat keparahannya. Setelah melewati fase pengelompokkan, data diserahkan kepada ahli untuk melakukan proses

validasi. Tahap ini dilakukan selama 2 hari. Data yang sudah divalidasi sesuai tingkat keparahan dimasukkan ke folder sesuai kategori masing masing lalu siap untuk diolah. Hasil pembagian ditampilkan di tabel 1. Pengolahan data dilakukan pada *Kaggle* dengan cara diupload sebagai dataset yang siap untuk diolah dalam proses klasifikasi pada citra.



Gambar 2. Citra penyakit leafblast (dari atas ke bawah) kategori sehat-ringan, sedang dan berat

Tabel 1. Pengelompokkan data sesuai kategori

No	Kategori	Jumlah
1	Sehat-Ringan	100
2	Sedang	100
3	Berat	100
Total		300

2.2. Tahap Pre-Processing

Setelah tahap pengumpulan data yang dimana pengambilan, pengelompokkan serta validasi dilakukan. Data kemudian diunggah ke *Kaggle* dengan menjadi dataset yang siap digunakan dalam penelitian ini. Setelah diunggah proses berlanjut ke tahap *Pre-Processing*, yang dimana di dalam tahap ini program

melakukan eksplorasi data dengan menampilkan citra sesuai dengan kategori ketika fase pengelompokkan.

Ketika eksplorasi data sudah dilakukan, *resize* citra dilakukan untuk menyesuaikan piksel dari citra dengan ukuran yang sudah disiapkan dari model *MobileNetV2* yang sudah disiapkan sekaligus dapat menghasilkan performa yang maksimal pada tahap pengujian dan validasi.

2.3. Tahap Processing

Penelitian ini menggunakan metode *Convolutional Neural Network* dengan arsitektur *MobileNetV2* yang dimana menerapkan metode Transfer Learning dalam pembuatan model yang siap pakai (*pre-trained model*) untuk klasifikasi citra tingkat keparahan penyakit *leafblast* pada tanaman padi. Peneliti menggunakan arsitektur ini karena terbukti untuk melakukan proses klasifikasi, segmentasi dan pengenalan pada objek[8]. Pada arsitektur *MobileNetV2* ada 2 fitur yang dimana sangatlah penting dan mengunggulkan arsitektur ini yaitu *Linear Bottleneck* dan *Shortcut Connection* antar *Bottleneck*. Kedua fitur ini menjadikan proses training pada *neural network* dapat berjalan dengan lebih efisien dan menghasilkan akurasi yang lebih baik[9].

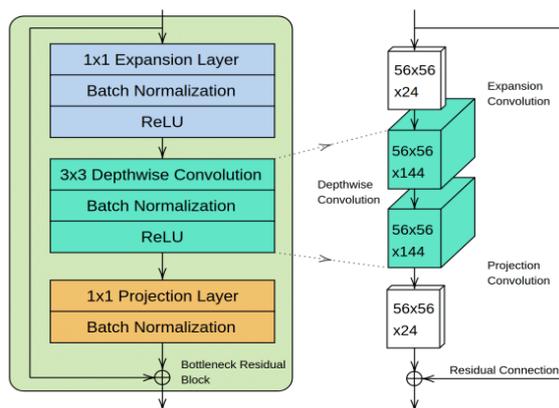
Pada penelitian ini, peneliti mengusulkan arsitektur model dengan input yang digunakan adalah gambar dari mode RGB 160X160 pixel dengan array 3 dimensi dan network yang sudah *pre-trained* oleh *imagenet*. Pada model ini terdiri dari 17 blok yang dimana berisikan beberapa *layers* yaitu *Expansion Layer*, *Depthwise Convolution*, *Projection Layer* yang dimana setiap *layer* terkecuali *Projection Layer* memiliki *Conv2D*, *BatchNormalization* dan *Relu* pada layernya kecuali *Projection Layer* yang tidak mempunyai *Relu*. Lalu setelah melalui blok dari *MobileNetV2* ditambahkan 1 Layer konvolusi, 1 Layer *BatchNormalization*, *Dense Layer* 256 dengan *activation 'relu'* yang dimana sudah dimasukkan *MobileNetV2* ke dalam layer ini, *Dense Layer* yang sama dengan sebelumnya, dan diakhiri dengan *Dense Layer* dengan output terdiri atas 3 kelas dengan *activation 'softmax'*. Adapun parameter yang digunakan ditampilkan di tabel 2.

Tabel 2. Parameter Arsitektur MobileNetV2

Parameter	Value
Bottleneck Residual	17
Activation Function	relu, softmax
Optimizer	adam
Loss Function	Categorical Crossentropy
Epoch	50
Batch Size	64

Hal yang diunggulkan dari model *MobileNetV2* adalah adanya *residual bottleneck* yang dirancang untuk klasifikasi gambar dan pembuatan fitur generik [10]. Layaknya *MobileNetV1*, Model ini

membagi konvolusi menjadi *Depthwise Convolution* dan *Pointwise Convolution*, akan tetapi di dalam *MobileNetV2* menambahkan dua fitur baru yaitu *Linear Bottleneck* dan *Shortcut Connection*. Pada bagian *bottleneck* terdapat input dan output antara model sedangkan lapisan atau *layer* bagian dalam mengenkapsulasi kemampuan model dalam mengubah input dari konsep yang rendah ke *descriptor* yang tinggi. Shortcut antar *bottlenecks* memungkinkan *Training* pada suatu model lebih cepat dan akurasi menjadi lebih baik[11]. Gambar 3 menampilkan bentuk secara menyeluruh mengenai arsitektur *MobileNetV2*.



Gambar 3. Arsitektur *MobileNetV2*

Batch Normalization diterapkan untuk memaksimalkan pelatihan dan meminimalisir kemungkinan terjadinya sebuah *overfitting*[12]. Sementara untuk penggunaan *Dropout* sudah terbukti dalam peningkatan akurasi dan juga dapat membantu menghindari *overfitting*[13] yang terjadi dalam klasifikasi citra di penelitian ini. *Optimizer* yang digunakan dalam penelitian ini adalah *Adam*. *Optimizer Adam* merupakan salah satu Teknik *optimizer* yang digunakan untuk memperbarui bobot pada model[14]. Adapun fungsi *loss* yang digunakan adalah *Categorical Crossentropy* untuk menerapkan sebuah Teknik untuk meminimalisir *loss* dan meningkatkan akurasi pada model[15].

Setelah dilakukan perancangan model *MobileNetV2*, peneliti membuat variabel *Callback*[16] yang akan diterapkan pada proses pelatihan model. *Callback* yang pertama memanfaatkan fungsi *ModelCheckpoint* dari Library *Tensorflow* pada modul *keras*. Fungsi tersebut untuk menyimpan model menjadi file pada setiap iterasi pelatihan (*Epoch*) yang dimana jika matrik *val_accuracy* mengalami peningkatan selama proses iterasi (*Epoch*) pada pelatihan model. Adapun *Callback* yang kedua menggunakan fungsi *EarlyStopping* dari library yang sama dengan *Callback* pertama. Fungsi *Early Stopping* digunakan untuk menghentikan proses pelatihan pada model apabila nilai pada metrik *val_loss* tidak mengalami perubahan selama 3 *epochs*. *ModelCheckpoint* dan *Early Stopping* merupakan 2 metode yang disediakan dan dikembangkan oleh

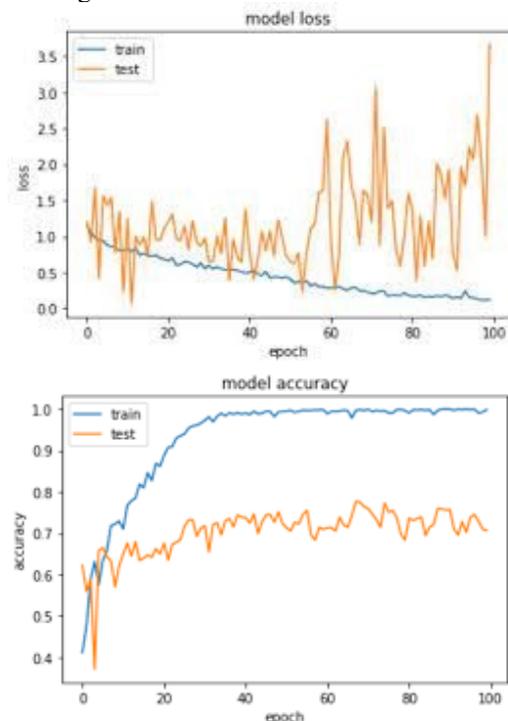
FastAI untuk melakukan 2 metode pelacakan yang dimana bekerja Ketika pelatihan dimulai dan memantau nilai metrik *loss validation* dengan learning rate yang telah ditentukan. Dengan menggunakan *Early Stopping* dapat mencegah terjadinya *overfitting*[14].

Peneliti melakukan tahap pelatihan pada model dengan melakukan *epochs* pada model. Pelatihan yang dilakukan dengan menggunakan sebanyak 50 *Epochs* pada pelatihannya dengan dataset. Adapun hasil yang diperoleh pada pelatihan model berupa hasil evaluasi terhadap data test dan validasi untuk melihat hasil kinerja setelah tahap pelatihan dilakukan

2.4. Tahap Uji Coba dan Validasi

Setelah dilakukan pelatihan pada model *MobileNetV2*, pada tahap ini dilakukan evaluasi terhadap model berupa grafik akurasi pelatihan, grafik *loss*, *Classification Report* dan *Confusion Matrix* yang didapat setelah selesai proses iterasi (*Epoch*) pelatihan.

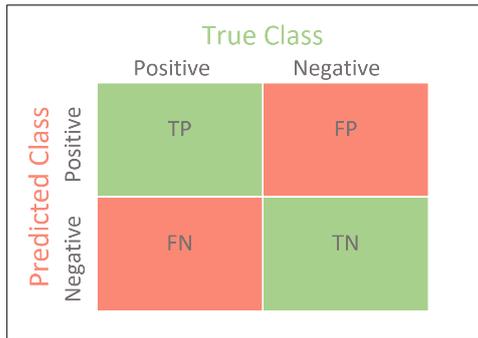
Untuk grafik akurasi pelatihan, nilai pada sumbu X menunjukkan jumlah epoch pelatihan yang dijalani selama proses. Sedangkan nilai pada sumbu Y menunjukkan nilai tingkat akurasi. Sedangkan grafik *loss* pelatihan, sumbu X menunjukkan *epoch* pelatihan yang telah dijalani oleh model dan sumbu Y menunjukkan nilai *loss* model selama pelatihan. Untuk contoh gambar grafik akurasi dan grafik *loss* bisa dilihat di gambar 4.



Gambar 4. Contoh Grafik *Loss* dan *Accuracy*

Selanjutnya, evaluasi terhadap model dilihat melalui tabel *Confusion Matrix* untuk mengukur kinerja metode *machine learning* dalam mengetahui seberapa banyak model mampu melakukan sebuah prediksi dengan benar untuk sesuai kategorinya[17].

Untuk contoh *Confusion Matrix* bisa dilihat di gambar 5.



Gambar 5. Contoh Confusion Matrix

Selain *Confusion Matrix*, hasil evaluasi dilihat menggunakan *Classification Report* yang dimana melihat besar akurasi prediksi yang benar dalam keseluruhan citra di dalam dataset. Jika akurasi mendekati angka 1 dapat diindikasikan bahwa model memiliki performa yang bagus. Jika mendekati angka 0 maka merepresentasikan sebaliknya[18]. Selain akurasi, terdapat juga di dalam *Classification Report* yaitu *precision*, *recall* dan *f1-score* yang mampu mengukur performa dari sebuah model. *Precision* adalah prediksi benar positif dengan keseluruhan hasil yang diprediksi positif. Sedangkan *recall* merupakan rasio prediksi benar positif dibandingkan dengan data keseluruhan yang benar positif. Adapun *f1-score* merupakan perbandingan rata-rata *precision* dan *recall* yang dibobotkan[19]. Untuk contoh *Classification Report* bisa dilihat di gambar 6.

	precision	recall	f1-score	support
Iris-setosa	1.00	1.00	1.00	50
Iris-versicolor	0.77	0.96	0.86	50
Iris-virginica	0.95	0.72	0.82	50
avg / total	0.91	0.89	0.89	150

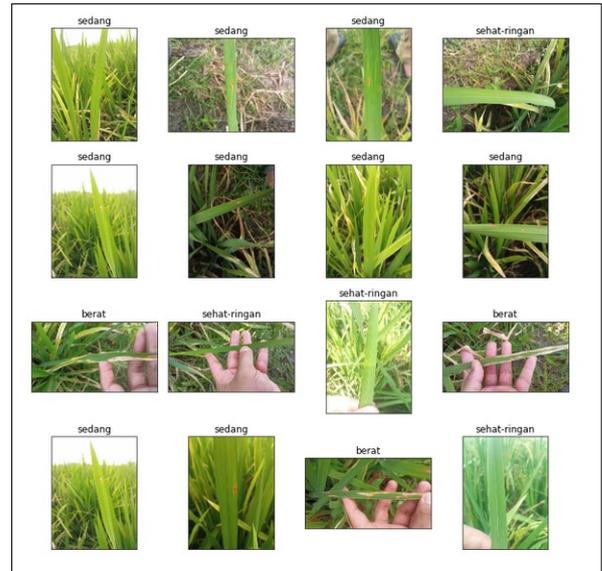
Gambar 6. Contoh Classification Report

3. HASIL DAN PEMBAHASAN

3.1. Tahap Pre-Processing

Pada bagian ini data yang sudah dikelompokkan ke masing-masing kategori tingkat keparahan dan sudah siap diolah yang dimana merupakan data yang sudah diunggah ke *Kaggle* dan berbentuk dataset. Program melakukan eksplorasi data untuk menentukan variabel per kategori pada seluruh dataset.

Data yang sudah di eksplorasi kemudian dilakkan proses *resize image* dengan resolusi 224 x 224 pixel. Proses ini berguna agar besar piksel gambar di dataset bisa menyamakan ukuran yang sudah disiapkan dari model *MobileNetV2* yang sudah disiapkan sekaligus dapat menghasilkan performa yang maksimal pada tahap pengujian dan validasi. Adapun gambar hasil observasi yang sudah ditentukan variabelnya bisa dilihat di gambar 7.



Gambar 7. Gambar dari dataset hasil observasi yang sudah ditentukan variabelnya sesuai kategori

3.2. Tahap Processing

Pada tahap ini, proses pelatihan model yang dilakukan dalam penelitian ini dieksekusi pada platform *Kaggle* dengan menggunakan *GPU P100* yang sudah disediakan. Penelitian ini melakukann uji coba klasifikasi penyakit daun padi berdasarkan tingkat keparahan. Setelah dilakukan tahap *pre-processing* data dibagi ke 3 macam kategori data yaitu data *training*, *testing* dan *validation*. Untuk pembagian datanya bisa dilihat di tabel 3.

Tabel 3. Pembagian DataCitra di tahap pre-processing

Nama Data	Sehat-Ringan	Sedang	Berat	Total
Data Train	64	64	64	192
Data Test	20	20	20	60
Data Validation	16	16	16	64

Setelah tahap tersebut dilakukan dilanjutkan proses augmentasi yang diberlakukan pada baik data *training*, data *testing* dan *validation*. Setelah itu dilakukan proses *resize image* dengan ukuran 224 X 224pixel dan diterapkan pada 3 kategori data tersebut dan *rescale layer*. *Rescale layer* bertujuan untuk konversi gambar yang bernilai dari [0 – 255] ke nilai [-1, 1][14].

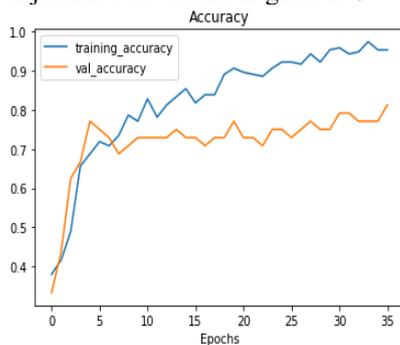
Setelah melakukan *resize image* dan *rescale layer*, proses klasifikasi dilakukan menggunakan arsitektur *Pre-Trained MobileNetV2* menggunakan 17 blok *Residual Bottleneck* yang terdiri dari beberapa *layers* yaitu *Expansion Layer*, *Depthwise Convolution*, *Projection Layer* yang dimana setiap *layer* terkecuali *Projection Layer* memiliki *Conv2D*, *BatchNormalization* dan *Relu* pada layernya kecuali *Projection Layer* yang tidak mempunyai *Relu*. Lalu

setelah melalui blok dari *MobileNetV2* ditambahkan 1-layer konvolusi, 1-layer *BatchNormalization*, *Dense Layer* 256 dengan *activation* 'relu' yang dimana sudah dimasukkan *MobileNetV2* ke dalam layer ini, *Dense Layer* yang sama dengan sebelumnya, dan diakhiri dengan *Dense Layer* dengan output terdiri atas 3 kelas dengan *activation* 'softmax'. Dari proses klasifikasi menggunakan arsitektur model ini, gambar dapat terklasifikasikan.

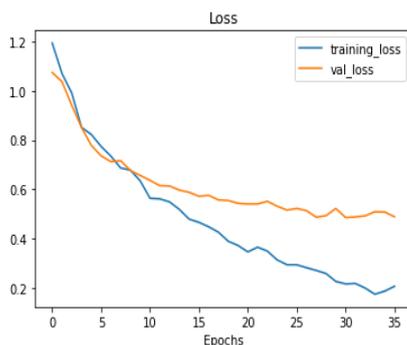
Berikutnya dibuat 2 variabel *callback* yang akan diterapkan pada proses pelatihan model. *Callback* yang pertama memanfaatkan fungsi *ModelCheckpoint* dari library *Tensorflow* pada modul *keras*. Fungsi tersebut untuk menyimpan model menjadi file pada setiap iterasi pelatihan (*Epoch*) yang dimana jika matrik *val_accuracy* mengalami peningkatan selama peningkatan selama proses iterasi pelatihan model. Adapun *Callback* yang kedua menggunakan fungsi *EarlyStopping* dari library yang sama pada *callback* pertama. Fungsi *EarlyStopping* digunakan untuk menghentikan proses pelatihan pada model apabila nilai pada metrik *val_loss* tidak mengalami perubahan selama 3 iterasi pelatihan (*Epochs*). Model kemudian dilatih dengan 50 iterasi pelatihan (*Epochs*) dan tidak lupa akan menerapkan kedua *callback* di atas.

3.3. Tahap Uji Coba dan Validasi

Pada pengujian model didapat hasil yang menunjukkan grafik akurasi pelatihan (*training accuracy*) dan akurasi validasi (*validation accuracy*) yang didapat setelah proses iterasi pelatihan (*Epoch*) terbilang cukup baik. Hal ini ditandai dengan hasil test akurasi mencapai 78.33% yang dimana belum bisa terlihat konvergen sampai tahap akhir iterasi pelatihan. Untuk penjelasan bisa dilihat di gambar 8.



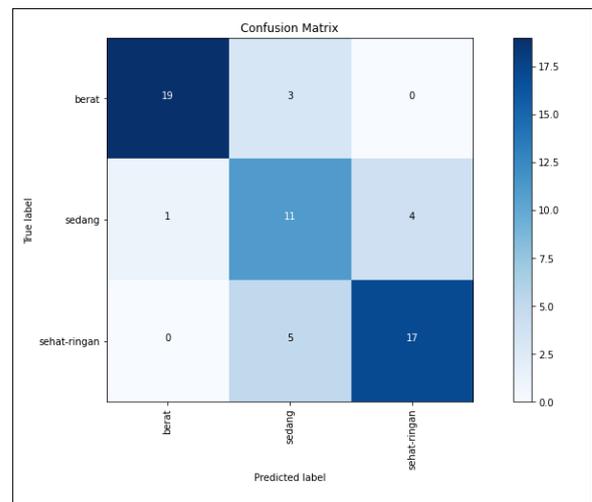
Gambar 8. Grafik Accuracy



Gambar 9. Grafik Loss

Adapun pada grafik loss yang didapati dari hasil pengujian sebesar 0.68% dengan rentang nilai dari 0 hingga 1,2. Adapun penjelasan bisa dilihat di gambar 9.

Selanjutnya, evaluasi terhadap model dilihat melalui tabel *confusion matrix* untuk mengukur kinerja mesin yang sudah diterapkan dengan model untuk mengetahui seberapa banyak model yang bisa melakukan sebuah prediksi dengan benar dan salah dari seluruh total keseluruhan data[14]. Adapun hasil dari *confusion matrix* dapat dilihat pada gambar 10. Berdasarkan data test gambar yang digunakan, didapati bahwa dari masing masing 60 data citra penyakit padi dengan kelas tingkat keparahan, memberikan hasil yang cukup bagus.



Gambar 10. Hasil Tabel Confusion Matrix

Adapun hasil evaluasi menggunakan *Classification Report* dapat dilihat pada tabel 11. Dilihat nilai akurasi model sebesar 0.78% berdasarkan nilai *accuracy* tertinggi. Selain akurasi terdapat *precision*, *recall* dan *f1-score* yang mampu mengukur sebuah performa pada model[14]. Di dalam pengujian ini hasil keseluruhan *classification report* dapat dilihat bahwa kelas sedang memiliki nilai yang rendah dibandingkan kelas sehat-ringan dan berat yaitu 0.58% untuk *precision*, 0.69% untuk *recall* dan 0.63% untuk *f1-score*.

Tabel 11. Classification Report

Nama Kategori	Precision	Recall	f1-score
Berat	0.950000	0.863636	0.904762
Sedang	0.578947	0.687500	0.628571
Sehat-Ringan	0.809524	0.772727	0.790698
Accuracy	0.783333		

Model *MobileNetV2* yang sudah dilatih kemudian dilakukan prediksi pada gambar secara acak dengan urutan 5 prediksi yaitu sedang, sedang, sehat-ringan,

berat, sehat-ringan. Adapun hasil prediksi gambar bisa dilihat di gambar 12. Dari gambar tersebut, masih ada prediksi yang gagal dan tidak sesuai dengan kategori sebenarnya.



Gambar 12. Hasil Prediksi citra menggunakan model MobileNetV2

4. KESIMPULAN

Berdasarkan seluruh rangkaian penelitian yang dilakukan, maka dapat ditarik beberapa kesimpulan. Penggunaan metode *Deep Convolutional Neural Network* dengan menggunakan *pre-trained MobileNetV2* untuk klasifikasi tingkat keparahan penyakit *leafblast* tanaman padi mendapatkan hasil cukup baik dengan memperoleh rata-rata valuasi akurasi sebesar 78.33%. Namun terdapat kekurangan yaitu masih tidaknya konvergen dalam proses iterasi data (*Epochs*) dan rendahnya pada kategori tingkat keparahan ‘sedang’ dalam keseluruhan classification report.

Hasil dari penelitian ini diharapkan terus dikembangkan khususnya dalam hal identifikasi yang sesuai dengan tingkat keparahan per kategorinya. Penelitian kedepannya diharapkan menggunakan jumlah kelas dan dataset lebih banyak sehingga klasifikasi tingkat keparahan penyakit tanaman padi ini semakin bervariasi, sesuai dengan perkembangan untuk kedepannya. Dengan begitu manfaat yang didapat dari hasil penelitian dapat dirasakan dampaknya bagi para petani dan seluruh masyarakat Indonesia.

5. DAFTAR PUSTAKA

- [1] S. Agustiani, Y. Tajul Arifin, A. Junaidi, S. Khotimatul Wildah, and A. Mustopa, “Klasifikasi Penyakit Daun Padi menggunakan Random Forest dan Color Histogram,” *J. Komputasi*, vol. 10, no. 1, 2022, doi: 10.23960/komputasi.v10i1.2961.
- [2] T. Suganda, E. Yulia, F. Widiyanti, and H. Hersanti, “Intensitas Penyakit Blas (*Pyricularia oryzae* Cav.) pada Padi Varietas Ciherang di Lokasi Endemik dan Pengaruhnya terhadap Kehilangan Hasil,” *Agrikultura*, vol. 27, no. 3, pp. 154–159, 2016, doi: 10.24198/agrikultura.v27i3.10878.
- [3] S. Indrayani, A. Nasution, and E. S. Mulyaningsih, “Analisis ketahanan padi gogo dan padi sawah (*Oryza sativa* L) terhadap empat ras penyakit blas (*Pyricularia grisea* Sacc),” *J. Agric.*, vol. 3, no. 1, pp. 53–62, 2013.
- [4] A. Nasution and B. Nuryanto, “Penyakit Blas *Pyricularia grisea* pada Tanaman Padi dan Strategi Pengendaliannya,” *Iptek Tanam. Pangan*, vol. 9, no. 2, pp. 85–96, 2015.
- [5] D. Eka Kusumawati and I. Istiqomah, “POTENSI AGENSIA HAYATI DALAM MENEKAN LAJU SERANGAN PENYAKIT BLAS (*Pyricularia oryzae*) PADA TANAMAN PADI,” *VIABEL J. Ilm. Ilmu-Ilmu Pertanian.*, vol. 14, no. 2, pp. 1–13, 2020, doi: 10.35457/viabel.v14i2.1235.
- [6] R. Indraswari, R. Rokhana, and W. Herulambang, “Melanoma image classification based on MobileNetV2 network,” *Procedia Comput. Sci.*, vol. 197, pp. 198–207, 2021, doi: 10.1016/j.procs.2021.12.132.
- [7] M. U. Hossain, M. A. Rahman, M. M. Islam, A. Akhter, M. A. Uddin, and B. K. Paul, “Automatic driver distraction detection using deep convolutional neural networks,” *Intell. Syst. with Appl.*, vol. 14, 2022, doi: 10.1016/j.iswa.2022.200075.
- [8] M. Toğaçar, Z. Cömert, and B. Ergen, “Intelligent skin cancer detection applying autoencoder, MobileNetV2 and spiking neural networks,” *Chaos, Solitons and Fractals*, vol. 144, p. 110714, 2021, doi: 10.1016/j.chaos.2021.110714.
- [9] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemant, “SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2,” *Sustain. Cities Soc.*, vol. 66, no. December 2020, p. 102692, 2021, doi: 10.1016/j.scs.2020.102692.
- [10] C. H. Karadal, M. C. Kaya, T. Tuncer, S. Dogan, and U. R. Acharya, “Automated classification of remote sensing images using multileveled MobileNetV2 and DWT techniques,” *Expert Syst. Appl.*, vol. 185, no. July, p. 115659, 2021, doi: 10.1016/j.eswa.2021.115659.
- [11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.
- [12] K. Thenmozhi and U. Srinivasulu Reddy, “Crop pest classification based on deep convolutional neural network and transfer learning,” *Comput. Electron. Agric.*, vol. 164, no. July, p. 104906, 2019, doi: 10.1016/j.compag.2019.104906.
- [13] C. R. Rahman *et al.*, “Identification and recognition of rice diseases and pests using

- convolutional neural networks,” *Biosyst. Eng.*, vol. 194, pp. 112–120, 2020, doi: 10.1016/j.biosystemseng.2020.03.020.
- [14] 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.
- [15] M. K. Priya and S. Dhanabal, “Analyses of Nine Different Types of Diseases in Paddy with Hybrid Algorithms using Deep Learning,” *IJERT-International J. Eng. Res. Technol.*, vol. 8, no. 08, pp. 1–7, 2020.
- [16] A. Chakraborty, D. Kumer, and K. Deeba, “Plant Leaf Disease Recognition Using Fastai Image Classification,” *Proc. - 5th Int. Conf. Comput. Methodol. Commun. ICCMC 2021*, no. Iccmc, pp. 1624–1630, 2021, doi: 10.1109/ICCMC51019.2021.9418042.
- [17] M. Koklu, I. Cinar, and Y. S. Taspinar, “Classification of rice varieties with deep learning methods,” *Comput. Electron. Agric.*, vol. 187, no. June, p. 106285, 2021, doi: 10.1016/j.compag.2021.106285.
- [18] S. Ramesh and D. Vydeki, “Application of machine learning in detection of blast disease in south indian rice crops,” *J. Phytol.*, vol. 11, pp. 31–37, 2019, doi: 10.25081/jp.2019.v11.5476.
- [19] A. Sagar and J. Dheebea, “On Using Transfer Learning For Plant Disease Detection,” *bioRxiv*, no. July, p. 2020.05.22.110957, 2020, doi: 10.13140/RG.2.2.12224.15360/1.